# VERTEX COLOURINGS OF EDGE-COLOURED GRAPHS

by

Richard Brewster
B.Sc. University of Victoria 1987
M.Sc. University of Victoria 1988

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
in the Department
of
Mathematics and Statistics


SIMON FRASER UNIVERSITY
April 1993

ISBN 0-315-91150-6

# APPROVAL

**Name:** Richard Brewster

**Degree:** Doctor of Philosophy

**Title of thesis:** Vertex Colourings of Edge-Coloured Graphs

**Examining Committee:** Dr. S. K. Thomason

Chair

---

Dr. P. Hell
Professor of Mathematics and Statistics
Senior Supervisor

---

Dr. B. Alspach
Professor of Mathematics and Statistics

---

Dr. K. Heinrich
Professor of Mathematics and Statistics

---

Dr. X. Zhu
Interal External Examiner
Visiting Professor of Computing Science

---

Dr. G. Sabidussi
Professor of Mathematics
Univeristy of Montreal
External Examiner

**Date Approved:** April 16, 1993

Title of Thesis/Project/Extended Essay

Vertex Colourings of Edge-Coloured Graphs

Author: ______________________

(signature)

Richard Brewster

(name)

April 8th /93

(date)

# Abstract

The point of departure of this thesis is the following classical vertex-colouring problem: Let $n$ be a fixed integer. Given a graph $G$, does $G$ admit an $n$-colouring (a mapping $f : V(G) \to \{1, 2, \ldots, n\}$ such that $f(g) \neq f(g')$ whenever $gg'$ is an edge of $G$)? A generalization of this problem which has attracted much attention recently is as follows: Let $H$ be a graph. An *$H$-colouring* of a graph $G$, or a homomorphism of $G$ to $H$, is a mapping $f : V(G) \to V(H)$ such that $uv \in E(G)$ implies $f(u)f(v) \in E(H)$. The *$H$-colouring problem* is: Given an input edge-coloured graph $G$, does there exist an $H$-colouring of $G$?

In this thesis we investigate the corresponding problem for a generalization of graphs: an *edge-coloured graph* is a relational system, $G = (V(G), E_1(G), E_2(G), \ldots, E_k(G))$, where $V(G)$ is a set of vertices and each $E_i(G)$ is a symmetric binary relation on $V(G)$. The elements of $E_i(G)$ are referred to as the edges of colour $i$.

We present some new constructions for studying the complexity of $H$-colouring for edge-coloured graphs. For the majority of the thesis we use these tools to classify the complexity of $H$-colouring where $H$ is a member of some particular family of edge-coloured graphs.

In the spirit of the previous work on $H$-colourings these complexity classifications typically depend on the existence (or lack of existence) of some structure in $H$. We

present evidence suggesting that for edge-coloured graphs a structural characterization that completely classifies which $H$-colouring problems are NP-complete and which are polynomial is unlikely. This is similar to the case of directed graphs. Indeed, we establish a polynomial equivalence between the complexity of $H$-colouring for biparite two-edge-coloured graphs and bipartite digraphs. We show that the problem is polynomial for paths and that there exists trees, cn as few as 12 vertices, for which the problem is NP-complete. We study the problem for cycles and present an infinite family of NP-complete cycles with two edge colours; moreover, any cycle smaller than the minimal element of the family is polynomial. We study the problem for cliques and completely classify the complexity for all cliques on three or fewer vertices with two edge colours and for all digon-free cliques on four vertices with two edge colours. We show that a clique with $k$ edge colours is NP-complete if it has more than $2^k$ vertices and that there exists cliques with $k$ edge colours and at most $2^k$ vertices which are polynomial.

We also establish an equivalence between $H$-colouring for edge-coloured graphs and a new homomorphism problem - the Sabidussi Homomorphism Problem and thereby we are able to classify the complexity for a large family of these problems.

# Acknowledgements

The author wishes to thank Pavol Hell for all his support during the writing of this thesis: academic and finacial. The author is also indebted to Gary MacGillivary for his many helpful suggestions and the "Regina Workshops". The construction used in Chapter Four was suggested by Joergen Bang-Jensen during the author's visit to Odense Univeristy. The author wishes to thank Dr. Bang-Jensen for the finacial support during the visit and for the helpful suggestions.

Thanks is also owed to the many others. Especially my parents, my fellow graduate students (including Susan "time for coffee" Marshall), the SFU Tri-Club, the Math Department at Capilano College, and Julia.

# Contents

# List of Figures

# Chapter 1

# Introduction

Graph colourings arise from a variety of applications including scheduling, combinatorial games, frequency assignments and others, [17], [23]. These applications have given rise to many generalizations of classical graph colourings. However, even the simplest graph colouring problems turn out to be very difficult to solve. Indeed, to test whether a given graph has a proper three-colouring is one of the basic NP-complete problems.

Suppose $G$ and $H$ are graphs. A ***homomorphism*** from $G$ to $H$, $f : G \to H$, is a mapping $f : V(G) \to V(H)$ such that if $uv$ is an edge of $G$, then $f(u)f(v)$ is an edge of $H$. If $G$ admits a homomorphism to $H$ we say $G$ is homomorphic to $H$ and write $G \to H$. If $G$ does not admit a homomorphism to $H$ we write $G \not\to H$. We say that $H$ is the "target". An ***$H$-colouring*** of $G$ is simply a homomorphism of $G$ to $H$. In particular, a graph $G$ admits a homomorphism to $K_n$ if and only if $G$ has a proper $n$-colouring. The study of graph homomorphisms has proven extremely powerful in the study of these generalized graph colourings. In fact, given that a homomorphism is a very natural mathematical object, some might argue that $H$-colouring is the *correct* way to examine these colouring problems.

The work in this thesis grew out of an attempt to translate colouring problems into homomorphism problems. In order to achieve this translation, a generalization of classical graphs had to be considered. The generalization is that of *edge-coloured graphs.* An *edge-coloured graph* is a relational system, $G = (V(G), E_1(G), E_2(G), \ldots, E_k(G))$, where $V(G)$ is a set of vertices and each $E_i(G)$ is a symmetric binary relation on $V(G)$; the elements of $E_i$ are called the edges of colour $i$. The number of edge sets, $k$, is called the *multiplicity* of $G$. Given two edge-coloured graphs $G$ and $H$ both of multiplicity $k$, a *homomorphism*, $f$, from $G$ to $H$ is a mapping $f : V(G) \rightarrow V(H)$ such that if $uv \in E_i(G)$, then $f(u)f(v) \in E_i(H)$ for each $i = 1, \ldots, k$.

For example, consider the following problem ("colouring with a condition at distance two") investigated by Griggs and Yeh [13]. Given a graph $G$, is there a function $f : V(G) \rightarrow \{1, 2, \ldots, n\}$ such that for all edges $uv$, $|f(u) - f(v)| \geq 2$ and for all pairs of vertices, $\{u, v\}$, at distance 2, $|f(u) - f(v)| \geq 1$. We can express this as a homomorphism problem. Given $G$ above, define $G'$ as:

- $V(G') = V(G)$,
- $E_1(G') = E(G)$, and
- $E_2(G') = \{uv | u, v \in V(G)$ and $u$ and $v$ are distance two apart in $G$ $\}$.

Define an edge-coloured graph $H$ as:

- $V(H) = \{1, 2, \ldots, n\}$,
- $E_1(H) = \{uv | u, v \in V(H); |u - v| \geq 2\}$,

- $E_2(H) = \{uv | u, v \in V(H); |u - v| \geq 1\}$.

It is easy to see that a function $f$ of the form above is just a homomorphism $G' \to H$.

The majority of this thesis concerns the following problem. Let $H$ be an edge-coloured graph. The $H$-colouring problem is the following:

**H-COL**

INSTANCE: An edge-coloured graph $G$.

QUESTION: Does there exist an $H$-colouring of $G$?

For the case of classical graphs, i.e., edge coloured-graphs of multiplicity one, the complexity of $H$-COL has been completely determined by Hell and Nešetřil [19]. They proved that $H$-COL is NP-complete if $H$ contains an odd cycle and is polynomial otherwise.

Many authors have studied the above problem when $H$ is a directed graph, but as yet no complete classification of $H$-COL exists nor has a conjecture about such a classification been presented. However, Bang-Jensen and Hell have a conjecture concerning a partial classification. In order to describe the conjecture we need the following definition. Suppose $H$ is a digraph. A homomorphism $r$ from $H$ to a subgraph $H'$ of $H$ is a *retraction* if $r$ is the identity map on $H'$. Bang-Jensen and Hell [2] have conjectured the following.

**Conjecture** Suppose $H$ is a digraph in which each vertex has in-degree at least one, and out-degree at least one. If $H$ does not admit a retraction to a directed cycle, then $H$-COL is NP-complete. Otherwise $H$-COL is polynomial.

This conjecture has been verified for many classes of digraphs in work by Bang-Jensen, Hell, and MacGillivray, [1], [2], [3], [24], [25].

In this thesis we attempt to identify those edge-coloured graphs $H$ for which $H$-COL is NP-complete, and those for which the problem is polynomial. In the spirit of the previous work on $H$-colourings these complexity classifications typically depend on the existence (or non-existence) of some structure in $H$. However, we present evidence suggesting that for edge-coloured graphs a structural characterization that completely classifies which $H$-colouring problems are NP-complete and which are polynomial is unlikely. This situation is similar to the case for directed graphs. Indeed, we establish a polynomial equivalence between the complexity of $H$-COL for bipartite edge-coloured graphs of multiplicity two, and bipartite digraphs.

In Chapter Two we present tools for studying $H$-COL. Some of these results are generalizations of the tools used for graphs and digraphs. This is the case, in particular, for the indicator construction and the subindicator construction. Other tools are new and their usefulness is truly only realized in the case of edge-coloured graphs; this is the case of the dominating loop lemma.

In Chapter Three we describe a problem similar to $H$-colouring based on a question asked by Sabidussi and Tardiff. Let $H$ and $Y$ be edge-coloured graphs and $h : H \to Y$ a homomorphism. The Homomorphism Factoring Problem is the following:

$HFP(H, h, Y)$

INSTANCE: An edge-coloured graph $G$ and a homomorphism $g : G \to Y$.

QUESTION: Does there exist a homomorphism $f : G \to H$ such that $h \circ f = g$?

We show that when $Y$ is a subgraph of $H$, and $h$ is the composition of a retraction of $H$ to $Y$ followed by an automorphism of $Y$, the problem $HFP(H,h,Y)$ is polynomial. We also show that for all graphs $Y$, with the exception of a four graphs $Y$, there exists a graph $H$ and a homomorphism $h : H \to Y$ such that $HFP(H,h,Y)$ is NP-complete. We also show that $HFP(H,h,Y)$ for graphs is equivalent to $H$-COL for edge-coloured directed graphs.

In Chapter Four we demonstrate an equivalence between $H$-COL for bipartite edge-coloured graphs of multiplicity two and $H$-COL for bipartite digraphs. Namely, let $H$ be a bipartite edge-coloured graph of multiplicity two. Then there exists a bipartite digraph, $D$, such that $H$-COL and $D$-COL are polynomially equivalent. We have a similar construction that begins with a bipartite digraph $D$. Namely, we can construct an edge-coloured graph $H$ such that $H$-COL and $D$-COL are polynomially equivalent. We use these constructions to both obtain new results for edge-coloured graphs (using digraphs) and conversely to obtain new results for digraphs (using edge-coloured graphs).

In Chapter Five we show that $H$-COL is polynomial when $H$ is an edge-coloured path. We also present edge-coloured trees for which $H$-COL is NP-complete. We present an "obstruction" type theorem for edge-coloured paths. Namely, if $G$ is an edge-coloured graph and $H$ is an edge-coloured path, then $G \not\to H$ if and only if there exists a path $P$ such that $P \to G$ and $P \not\to H$.

In Chapter Six we study $H$-COL for cycles. A *piece* of an edge-coloured cycle is a maximal monochromatic path. (Note for edge-coloured cycles with two edge-colours, the number of pieces is always even.) We show that in this case the problem is polynomial if the cycle has two or four pieces. Let $n \geq 6$. If $n \equiv 0 \pmod 4$, then there exists a cycle with two edge-colours and $n$ pieces for which $H$-COL is polynomial. If $n \equiv 2 \pmod 4$, then there exists a cycle with two edge-colours and $n$ pieces for which $H$-COL is NP-complete. For cycles with two edge-colours, we show that the problem is polynomial if all the pieces have the same parity. For the general case of $k$ edge-colours if all the pieces have odd length or if exactly one of the pieces has even length, we show the problem is polynomial. However, even for two edge-colours, the classification of the complexity of edge-coloured cycles remains an interesting open problem.

In Chapter Seven we study edge-coloured cliques, possibly including loops. In particular, we classify the complexity of $H$-colouring by an edge-coloured clique for all cliques with fewer than three vertices. We also classify the complexity of $H$-COL for cliques on three vertices with multiplicity two and all digon-free cliques on four vertices with multiplicity two. We present two infinite families of cliques for which the problem is polynomial. In fact, we show that for a clique $H$ of multiplicity $k$, $H$-COL is NP-complete if $H$ has more than $2^k$ vertices, and that for each $n \leq 2^k$, there exists a clique $H$ with $n$ vertices such that $H$-COL is polynomial.

Finally, in Chapter Eight we return to the case of graphs and digraphs. We study two problems. The first is to classify those graphs that contain, as subgraphs, all of their homomorphic images. We present a classification of such graphs. The second problem involves digraphs. Given a digraph $H$ and a digraph $G$, what is the smallest power, $k$, of $H$ such that $G$ admits a homomorphism to $H^k$? We study the complexity of this problem and show it is polynomial if $H$ is an oriented path but is NP-complete even for directed cycles.

## 1.1 Definitions and Preliminaries

We assume the reader is familiar with the basic notions and definitions of graph theory. We state below all definitions unique to this thesis as well as the common definitions and notations used. When not mentioned below, we use the notation and definitions of [5].

### 1.1.1 Basic Definitions

An *edge-coloured graph* is a *relational system*, $G = (V(G), E_1(G), E_2(G), \ldots, E_k(G))$, where $V(G)$ is the set of vertices and $E_1(G), E_2(G), \ldots, E_k(G)$ are symmetric binary relations on $V(G)$; the elements of $E_i$ are called the edges of colour $i$. The number of edge sets $k$ is called the *multiplicity of G*. An *edge-coloured directed graph* (or *edge-coloured digraph*) is just a relational system on a set of vertices $V(G)$ where each $E_i$ is a binary relation (which is not required to be symmetric). In this thesis we will restrict our attention to edge-coloured graphs. However, some results naturally generalize to the more general case of edge-coloured digraphs. These results are stated with the explicit use of the phrase edge-coloured digraph. Also, we reserve the use of the words graph and digraph to the case of multiplicity one; that is, to their classical usage. If we wish to explicitly state the number of edge colours in an edge-coloured graph we use the term *k-edge-coloured graph*. In particular, the term two-edge-coloured graph refers to an edge-coloured graph with $k = 2$. In the following assume $G$ and $H$ are edge-coloured graphs. Also suppose that red and blue are edge colours.

To maintain our analogy to undirected graphs, we shall identify pairs of opposite edges, i.e., for any edge-colour $i$ and pair $uv \in E_i(G)$ with $u \neq v$, we identify $uv$ and $vu$. Thus we consider $uv$ as one undirected edge. Since this is equivalent to viewing

$uv$ as a two-element subset, we sometimes write $\{u, v\}$ for $uv$.

Suppose $uv$ is a member of $E_i(G)$. We say $uv$ is an edge of colour $i$. We also say $u$ is *adjacent* to $v$ in $i$ or $u$ is *joined* to $v$ in $i$. The edge $uv$ is said to be *incident* with $u$ and $v$ and each of $u$ and $v$ is said to be *incident* with the edge $uv$. The vertex $u$ is said to be a *neighbour of colour* $i$ of $v$ or simply an $i$ *neighbour* of $v$. We let $N_i(v)$ denote the set of $i$ neighbours of $v$. The vertices $u$ and $v$ are called the *ends* of the edge $uv$.

A vertex incident with only blue edges is called a *blue only* vertex. A vertex incident with edges of at least two different colours is called a *mixed* vertex.

The *underlying graph* of $G$ is the graph on vertex-set $V(G)$ and edge-set $E(G)$ defined by: $uv \in E(G)$ if and only if $uv \in E_i(G)$ for some $i$. In other words, $E(G) = E_1(G) \cup \cdots \cup E_k(G)$. The use of $E(G)$ is common throughout the thesis, and we do not usually explicitly remind the reader that $E(G) = E_1(G) \cup \cdots \cup E_k(G)$.

We say $u$ is a *neighbour* of $v$ in $G$ if $u$ is a neighbour of $v$ in the underlying graph of $G$. We denote the neighbours of $v$ by $N(v)$. A similar remark applies to terms like "adjacent" and "joined to".

If $u$ and $v$ are both blue neighbours and red neighbours, we say $uv$ is a *digon*. We will use the term *red-blue-digon* if we wish to explicitly state the edge colours.

Suppose $uu$ is an edge of $G$. We call such an edge a *loop*. In the study of $H$-COL for graphs and digraphs, the existence of loops makes the problem trivial. If $H$ contains a loop at $v$, then any graph $G$ admits a homomorphism to $H$ simply by mapping all the vertices of $G$ to $v$. On the other hand, if $G$ contains a loop and admits a homomorphism to $H$, then $H$ must contain a loop. For the case of edge-coloured graphs, the existence of loops no longer necessarily makes the problem trivial. Therefore, in general we allow loops, although for simplicity we often restrict our attention to the loop-free case. In any section where we allow loops, we state this explicitly at the beginning of the section. Having said that, we observe that a vertex with a loop of every edge colour again makes the problem trivial for the same reason as above. To avoid trivialities, we never allow the existence of a vertex with a loop of every edge colour.

A *path of length* $n$ in $G$, denoted $P_n$, is a sequence of distinct vertices $v_0v_1v_2\ldots,v_n$ such that for each $i$, $1 \leq i \leq n-1$, $v_iv_{i+1} \in E(G)$. In other words, a path in $G$ is a path in the underlying graph of $G$. A path is called a *blue path* or a *path of colour blue* if each edge in the path is a blue edge. A *walk of length* $n$ is a sequence of vertices $w_0w_1w_2\ldots,w_n$ such that for each $i$, $1 \leq i \leq n-1$, $w_iw_{i+1} \in E(G)$.

Given vertices $u$ and $v$ in $G$, the *distance* between $u$ and $v$, denoted $d(u,v)$, is the length of a shortest path from $u$ to $v$. The distance in colour $i$, denoted $d_i(u,v)$, is the length of a shortest path of colour $i$ from $u$ to $v$. If there is no path (respectively no path of colour $i$) from $u$ to $v$, then $d(u,v) = \infty$ (respectively $d_i(u,v) = \infty$).

A *cycle of length* $n$, $C_n$, is a sequence of distinct vertices $v_0v_1v_2\ldots,v_{n-1}$ such that for each $i$, $1 \leq i \leq n-1$, $v_iv_{i+1} \in E(G)$ and $v_{n-1}v_0 \in E(G)$.

A subgraph $G'$ of $G$, denoted $G' \subseteq G$, is an edge-coloured graph where $V(G') \subseteq V(G)$ and $E_i(G') \subseteq E_i(G)$ for each edge-colour $i$. A subgraph $G'$ of $G$ is an *induced* *subgraph* if for each pair of vertices $u$ and $v$ in $V(G')$ we have $uv \in E_i(G')$ if and only if $uv \in E_i(G)$. We say $G'$ is *induced* by the set of vertices $V(G')$. The *spanning blue subgraph* of $G$ is the graph (multiplicity one) $(V(G), E_{blue}(G))$.

We say $G$ is *connected* if for each pair of vertices $u$ and $v$ in $G$, there exists a path from $u$ to $v$. We say $G$ is *connected in blue* if the spanning blue subgraph of $G$ is connected. A *component* of $G$ is a maximal connected subgraph of $G$. A *blue component* of $G$ is a maximal connected subgraph in the blue spanning subgraph of $G$.

A *clique* is an edge-coloured graph $H$ such that for each pair of distinct vertices $u$ and $v$ in $H$, $uv \in E(H)$. We say $H$ is a *blue clique* if the blue spanning subgraph of $H$ is a clique. Note that a blue clique may have some extra edges of other colours. A clique with $n$ vertices is called an $n$-clique. In particular a two-clique has two vertices. Note we reserve the use of the symbol $K_n$ for a clique with multiplicity one.

A set of vertices $S$ in $G$ is an *independent set* if for each $u$ and $v$ in $S$, $u$ and $v$ are non-adjacent in the underlying graph of $G$. (Hence, a single vertex with a loop is not an independent set.) We say $G$ is bipartite if the vertices of $G$ can be partitioned into two independent sets $G_0$ and $G_1$. We denote this partition by $(G_0, G_1)$. We say $G$ is a complete bipartite edge-coloured graph if $G$ is bipartite, with bipartition $(G_0, G_1)$, and for each $u \in G_0$ and $v \in G_1$, $uv \in E(G)$.

Let $G$ and $H$ be two disjoint edge-coloured graphs with the same multiplicity. The *union* of $G$ and $H$, denoted $G \cup H$, is the edge-coloured graph (also of multiplicity $k$) with vertex-set $V(G \cup H) = V(G) \cup V(H)$ and edge-sets $E_i(G \cup H) = E_i(G) \cup E_i(H)$

for $0 \leq i \leq k$.

Let $S$ be a set of edge-colours. Let $G$ be an edge-coloured graph with edge-colours $X \subseteq S$. Let $H$ be an edge-coloured graph with edge-colours $Y \subseteq S$. Thus, $G$ has multiplicity $|X|$ and $H$ has multiplicity $|Y|$. Let $Z \subseteq S$. The *join* of $G$ and $H$ with respect to $Z$ is the edge-coloured graph of multiplicity $|X \cup Y \cup Z|$ obtained by taking a copy of $G$ and a copy of $H$ and adding the edge $uv$ to $E_i(G \cup H)$ for each $u \in V(G)$, each $v \in V(H)$, and each $i \in Z$.

As mentioned above, the primary focus of this thesis is edge-coloured graphs, but there are times when we will use the term edge-coloured digraph. The edge $uv$ in a digraph is also called an *arc*. We refer to $u$ as an *in-neighbour* of $v$ and $v$ as an *out-neighbour* of $u$. An *oriented graph* is a digraph with no pair of vertices $u$ and $v$ such that both $uv$ and $vu$ are arcs. A *directed path* $v_0v_1 \ldots, v_n$ is a path where each $v_iv_{i+1}$, $0 \leq i \leq n-1$, is an arc. An *oriented path* $v_0v_1 \ldots, v_n$ is a path in which for all $0 \leq i \leq n-1$ either $v_iv_{i+1}$ or $v_{i+1}v_i$. Given a digraph $H$, the $k^{th}$ *power* of $H$, denoted $H^k$, is the digraph with vertex-set $V(H)$ and edge-set $uv \in E(H^k)$ if and only if there is a directed path from $u$ to $v$ in $H$ of length at most $k$.

### 1.1.2 Homomorphisms

We have already defined a homomorphism above. An *isomorphism*, $f$, from $G$ to $H$ is a homomorphism from $G$ to $H$ such that $f$ is one-to-one, onto, and $uv \in E_i(G)$ *if and only if* $f(u)f(v) \in E_i(H)$ for every edge-colour $i$. An *automorphism* of $G$ is an isomorphism of $G$ to itself.

Let $H$ be a subgraph of $G$. A *retraction*, $r : G \to H$, is a homomorphism that is the identity map on $H$. We say that $H$ is a *retract* of $G$. We say that $G$ is a *core* if there is no homomorphism from $G$ to a proper subgraph of $G$. Let $G$ be a finite edge-coloured graph. It is easy to prove that $G$ contains a subgraph $H$ that is a core and that there is a retraction $r : G \to H$. Moreover, $H$ is unique up to isomorphism. We call $H$ *the* core of $G$. This is proved for graphs in [19].

Suppose $H'$ is the core of $H$ and $r : H \to H'$ is a homomorphism of $H$ to $H'$. We have the inclusion mapping $i : H' \to H$ which is a homomorphism. Hence, $G \to H$ if and only if $G \to H'$. Thus when studying $H$-colouring we can restrict our attention to the case when $H$ is a core.

A final definition we require, the *product* of $G$ and $H$, denoted $G \times H$, is the edge-coloured graph with vertex-set $V(G) \times V(H)$ where $(g_1, h_1)(g_2, h_2) \in E_i(G \times H)$ if and only if $g_1 g_2 \in E_i(G)$ and $h_1 h_2 \in E_i(H)$. Note that $G \times H \to G$ and $G \times H \to H$ via the projections $\phi_G$ and $\phi_H$, where $\phi_G(g, h) = g$ and $\phi_H(g, h) = h$.

### 1.1.3 Congruences

Consider the path of length three, $P_3$. The graph that results when the end points of the path are identified is $K_3$. In fact, the identification *is*, in some sense, a homomorphism from $P_3$ to $K_3$. Many times we will want to talk about homomorphisms that result from certain identifications of vertices. There is a problem in that the target of this action is not defined until the identification is performed. However, a homomorphism can not be defined without stating the target. Therefore, we introduce a subject closely related to homomorphisms that will allow us to more rigorously define such identifications. Furthermore, the definition of this identification will define the target and implicitly define a homomorphism.

Let $H$ be an edge-coloured graph. We define a *congruence* $C$ on $H$ as a partition of the vertices into sets $S_0, S_1, \ldots, S_m$. The *quotient* of $C$, say $K$, is the edge-coloured graph on vertices $\{0, 1, \ldots, m\}$ with edge sets $E_t = \{ij|$ there is an edge of colour $t$ from some vertex in $S_i$ to some vertex in $S_j$ $\}$. Observe that the quotient is the edge-coloured graph obtained by contracting each $S_i$ to a single vertex. Also observe that there is a natural homomorphism $h : H \to K$ defined by:

$$h(v) = i \text{ if and only if } v \in S_i$$

In fact, we say the congruence $C$ *induces* the homomorphism $h$. Conversely, suppose that $f : G \to H$ is a homomorphism such that $f$ is onto the vertices of $H$ and such that for all $t$ and all $xy \in E_t(H)$ there exists $u$ and $v$ in $G$ so that $f(u) = x$ and $f(v) = y$. Then we can define a congruence on $G$ such that $H$ is the quotient of the congruence. Namely, if we label the vertices of $H$ with $0, 1, \ldots, m$, then:

$$S_i = f^{-1}(i) \text{ for } 0 \leq i \leq m,$$

is a congruence on $G$ with quotient equal to $H$. We say $H$ is a *homomorphic image* of $G$. We also denote $H$ as $f(G)$.

We now return to the example above. Given the path $P_3$, we wish to identify the end-vertices of the path. Define the congruence $C$ with the following three classes $S_0, S_1, S_2$. Let $S_0$ contain the two end points of the path. Let $S_1$ and $S_2$ each contain an interior vertex of $P_3$. The quotient of the congruence is $K_3$. The homomorphism induced by the congruence is exactly the one we wished to define.

### 1.1.4 Some Basic Results on Homomorphisms

**Lemma 1.1.1** *Let $G$ be a connected edge-coloured graph and $f : G \to H$ a homomorphism of $G$ to some edge-coloured graph $H$. Then $f(G)$ is connected.*

**Proof.** Let $x$ and $y$ be two vertices in $f(G)$. By definition of $f(G)$, there exists $u$ and $v$ in $G$ such that $f(u) = x$ and $f(v) = y$. Since $G$ is connected, there exists a path $(u = p_0)p_1 \ldots (p_n = v)$ such that $p_i p_{i+1} \in E(G)$ for $i = 0, 1, \ldots, n-1$. This implies $f(p_i)f(p_{i+1}) \in E(H)$ for $0 \le i \le n-1$, since $f$ is a homomorphism. Hence $(x = f(u) = f(p_0))f(p_1)\ldots(f(p_n) = f(v) = y)$ is a walk in $H$ containing a path from $x$ to $y$. ∎

**Lemma 1.1.2** *Let $G$ and $H$ be edge-coloured graphs. Then $G \to G \times H$ if and only if $G \to H$.*

**Proof.** If $G \to G \times H$, then from $G \times H \to H$ we have $G \to H$ by composition. On the other hand, suppose there exists a homomorphism $f : G \to H$. Let $\phi : G \to G \times H$ be the mapping defined by $\phi(g) = (g, f(g))$ for all $g \in V(G)$. Now if $g_1 g_2 \in E_i(G)$, then $(g_1, f(g_1))(g_2, f(g_2)) \in E_i(G \times H)$; thus, $\phi$ is indeed a homomorphism of $G$ to $G \times H$.∎

By examining the mapping $\phi$, we obtain the following corollary.

**Corollary 1.1.3** *Suppose $G$ and $H$ are edge-coloured graphs. If $G \to H$, then there exists a one to one homomorphism $\phi : G \to G \times H$ of the form $\phi(g) = (g, f(g))$ for all $g \in V(G)$.*

**Proof.** Using $\phi$ from the proof above, if $\phi(g_1) = \phi(g_2)$, then $(g_1, f(g_1)) = (g_2, f(g_2))$. This implies $g_1 = g_2$. ∎

Usings lemmas 1.1.1 and 1.1.2 we deduce the following.

**Corollary 1.1.4** *Suppose $G$ and $H$ are edge-coloured graphs. If $G$ is connected, then $G \to H$ if and only if there is a one to one homomorphism from $G$ to some connected component of $G \times H$ of the form $\phi(g) = (g, f(g))$ for all $g \in V(G)$.*

### 1.1.5 Complexity Theory

The following brief summary highlights the important details of complexity theory we require in this thesis. See [12] for a detailed treatment of the subject.

In the study of computational complexity attention is often restricted to decision problems. We use the terminology and the notation of [12] and briefly outline the main ideas below. A decision problem is a problem with only two possible solutions – YES and NO. A decision problem $\Pi$ consists of a set $D_\Pi$ of *instances* and a subset $Y_\Pi \subseteq D_\Pi$ of YES-instances. We describe a decision problem by a description of a generic-instance, $D$, for example, the edge-coloured graph $G$ in $H$-COL, and a question whose answer is YES if and only if $D \in Y_\Pi$. An algorithm *solves* a decision problem by computing whether or not a given instance is a YES-instance.

The *complexity* of an algorithm is a function, $f(n)$, from the size of the instance to the number of computational steps required to solve the problem. Here $n$ is some *reasonable* measure of the size of the instance. See [12] for more detail. If there is a polynomial $p(n)$ such that a given algorithm has complexity $O(p(n))$, then we say the algorithm is a *polynomial time algorithm.* We denote, by **P**, the set of decision problems that are solvable in polynomial time.

Let $\Pi_1$ and $\Pi_2$ be two decision problems. We say $\Pi_1$ *polynomially transforms* or *reduces* to $\Pi_2$ if there exists a function, $f$, from $D_{\Pi_1}$ to $D_{\Pi_2}$ such that:

- $f$ is computable in polynomial time, and
- for $I \in D_{\Pi_1}$, $I \in Y_{\Pi_1}$ if and only if $f(I) \in Y_{\Pi_2}$.

If $\Pi_1$ admits a polynomial transformation to $\Pi_2$, we write $\Pi_1 \alpha \Pi_2$. Observe that if $\Pi_2 \in \mathbf{P}$ and $\Pi_1 \alpha \Pi_2$, then $\Pi_1 \in \mathbf{P}$.

We denote by **NP** the set of decision problems that are solvable in polynomial time by a non-deterministic algorithm. See [12] for an explanation of non-determinism. One of the great open problems in complexity theory is whether or not **P=NP**.

A decision problem, $\Pi$, is NP-complete if:

- $\Pi \in \mathbf{NP}$.
- for all $\Sigma \in \mathbf{NP}$, $\Sigma \alpha \Pi$.

To show that a particular problem, $\Pi$, is NP-complete, we need to show first that $\Pi \in \mathbf{NP}$. It is easy to see the $H$-COL is in **NP**. Second, we choose some known NP-complete problem $\Sigma$ and show $\Sigma \alpha \Pi$. Since $\Sigma$ is NP-complete, we know that any problem in **NP** polynomially transforms to $\Sigma$. Composing this transformation with $\alpha$, we see that any problem in **NP** polynomially transforms to $\Pi$.

Suppose $\Pi$ and $\Sigma$ are two decision problems. A *polynomial time Turing reduction* of $\Pi$ to $\Sigma$ is a function $f$ from $D_\Pi$ to the power set of $D_\Sigma$ such that:

- $f$ can be computed in polynomial time, and

- for each $I \in D_\Pi$, $I \in Y_\Pi$ if and only if there is an $I' \in f(I)$ such that $I' \in Y_\Sigma$.

We write $\Pi \alpha_T \Sigma$ and say $\Pi$ *Turing reduces* to $\Sigma$.

If $\Sigma$ is in **P** and $\Pi \alpha_T \Sigma$, then $\Pi \in$ **P**. We say $\Pi$ is NP-hard if there exists an NP-complete problem $\Sigma$ such that $\Sigma \alpha_T \Pi$.

We conclude this section with an observation that allows us to assume for an $H$-colouring problem, $H$ is connected. We have already pointed out that we may assume that $H$ is a core.

**Proposition 1.1.5** *Let $H$ be an edge-coloured graph which is a core and let $H_1$ be a component of $H$. Then $H_1$-COL $\alpha_T$ $H$-COL.*

**Proof** Let $G$ be an instance of $H_1$-COL. Since $H$ is a core, $H_1$ does not admit a homomorphism to any other component of $H$. Let $h_1, h_2, \ldots, h_n$ be an enumeration of the vertices of $H_1$ and let $g$ be a vertex of $G$. Define $G_i$ to be the edge-coloured graph obtained by taking a copy of $H_1$ and a copy of $G$ and identifying $g$ and $h_i$. It is easy to check that $G \to H_1$ if and only if there exists an $i$ so that $G_i \to H$. ∎

## 1.2 Previous Work

The $H$-colouring problem for graphs and digraphs has received much recent attention, [19], [1], [2], [3], [14], [15], [25], [24]. In this section we present a brief survey of some of these results. The complexity of the $H$-colouring problem is completely determined for graphs by the following result of Hell and Nešetřil [19]:

**Theorem 1.2.1** *Let $H$ be a fixed graph. If $H$ contains an odd cycle, then $H$-COL is NP-complete. Otherwise, $H$ is bipartite and $H$-COL is polynomial.*

Since this completely classifies the problem for graphs, attention has since shifted to digraphs.

In their 1981 paper, Maurer, Sudborough, and Welzl [27] classify the complexity of $H$-COL for all three-vertex digraphs. This work was extended by Gutjahr [15] who classified the complexity of $H$-COL for all four-vertex digraphs.

Several families of polynomial digraphs were also presented in [27]. In particular, the authors show that $H$-COL is polynomial when $H$ is

- a directed path,
- a directed cycle, and
- a transitive tournament.

We have already presented a conjecture by Bang-Jensen and Hell concerning the complexity of $H$-COL for digraphs. This conjecture has been verified for many families of digraphs. In their 1990 paper, Bang-Jensen and Hell [2] verify the conjecture for digraphs consisting of two cycles and for complete bipartite digraphs. Bang-Jensen, Hell, and MacGillivray [1] have verified the conjecture for semicomplete digraphs (superdigraphs of tournaments). The conjecture has also been verified by MacGillivray [24] for vertex-transitive and arc-transitive graphs.

The concept of "hereditary hardness", [4], has been studied by Bang-Jensen, Hell, and MacGillivray. They present hereditarily hard digraphs in the sense that any digraph $H$ that contains a hereditarily hard digraph as a subgraph has the property that $H$-COL is NP-complete. They use this concept to study the above conjecture of

Bang-Jensen and Hell. Furthermore, they show the equivalence of this conjecture to a simpler conjecture.

The classification of $H$-COL for graphs in [19] implies essentially the only $H$-colouring problem for graphs that is polynomial is $K_2$-COL. Recall we can restrict our attention to the case when $H$ is a core. The algorithm for this problem is trivial. For oriented graphs we have polynomial algorithms that are no longer trivial or obvious. Classifying the complexity of $H$-COL for oriented paths proved difficult. In their 1992 paper, Gutjahr, Welzl, and Woeginger [14] defined an ***$\underline{X}$-graph*** to be a digraph for which there is an enumeration $v_1, v_2, \ldots, v_n$ of the vertices such that if $v_i v_j$ and $v_k v_l$ are arcs, then so is $v_{\min\{i,k\}} v_{\min\{j,l\}}$. The main result of their paper is as follows.

**Theorem 1.2.2** *Let $H$ be an $\underline{X}$-graph. Then $H$-COL is polynomial.*

It is easy to see that every oriented path is an $\underline{X}$-graph. Hence, if $H$ is an oriented path, then $H$-COL is polynomial.

In the same paper, the authors present an oriented tree, $T$ (with 287 vertices), such that $T$-COL is NP-complete.

The classification of oriented cycles has also proved difficult. Define an ***balanced oriented cycle*** to a be a cycle in which the number of forward arcs equals the number of backward arcs. Define an ***unbalanced oriented cycle*** to be a cycle which is not balanced. Gutjahr [15] and independently Zhu [30] have shown the following.

**Theorem 1.2.3** *Let $H$ be an unbalanced oriented cycle. Then $H$-COL is polynomial.*

Gutjahr, [15], has also constructed a balanced oriented cycle for which $H$-COL is NP-complete. These results suggest the complete classification of $H$-COL even for oriented trees and oriented cycles may prove difficult.

As mentioned above, the only polynomial $H$-colouring problem for graphs is $K_2$-COL. A classical result of graph theory is that a graph is bipartite if and only if it contains no odd cycle. We can rephrase this in terms of homomorphisms as follows. Let $G$ be a graph. There exists an odd cycle $C$ such that $C \to G$ if and only if $G \not\to K_2$.

Similar "obstruction" type results for homomorphisms to directed paths and directed cycles have been obtained by Häggkvist, Hell, Miller, Neumann-Lara [16]. These results have been extended to oriented paths by Hell and Zhu [20] as follows:

**Theorem 1.2.4** *Let $G$ be an oriented graph and $H$ an oriented path. Then there exists a path $W$ such that $W \to G$ and $W \not\to H$ if and only if $G \not\to H$.*

An oriented cycle, $C$, is called *nice* if for any oriented graph $G$, there exists a cycle $C'$ such that $C' \to G$ and $C' \not\to C$ if and only if $G \not\to C$. Hell, Zhou, and Zhu have shown that all unbalanced oriented cycles are nice and there are balanced oriented cycles that are not nice.

# Chapter 2

# $H$-Colouring Tools

## 2.1 Indicator Constructions

Both the indicator construction and the subindicator construction have proved very useful in the study of homomorphisms; see [19] and [25]. We extend these constructions to edge-coloured graphs. These will be used in Chapters Three, Five, Six, Seven, and Eight.

### 2.1.1 The Indicator Construction

Let $I_1, I_2, \ldots I_m$ be $m$ fixed edge-coloured graphs. For each $I_t$, $t = 1, \ldots, m$, let $i_t$ and $j_t$ be distinct vertices of $I_t$ such that there exists an automorphism, $\sigma_t$, of $I_t$ with $\sigma_t(i_t) = j_t$ and $\sigma_t(j_t) = i_t$. The *indicator construction* (with respect to $(I_1, i_1, j_1), (I_2, i_2, j_2), \ldots, (I_k, i_k, j_k)$) transforms a given edge-coloured graph $H$ into the following edge-coloured graph $H^*$. The edge-coloured graph $H^*$ has the same vertex set as $H$. The edge $hh' \in E_t(H^*)$ if and only if there is a homomorphism, $f : I_t \to H$, such that $f(i_t) = h$ and $f(j_t) = h'$. We now show that $H^*$ is an edge-coloured graph, i.e. each relation $E_t(H^*)$ is symmetric. Suppose there is a homomorphism, $f : I_t \to H$, such that $f(i_t) = h$ and $f(j_t) = h'$. That is, $hh' \in E_t(H^*)$. Recall there exists an automorphism, $\sigma_t$, of $I_t$ such that $\sigma_t(i_t) = j_t$ and $\sigma_t(j_t) = i_t$. Composing $f$ with $\sigma_t$ we get the homomorphism $f \circ \sigma_t : I_t \to H$ with $f \circ \sigma(i_t) = h'$ and $f \circ \sigma(j_t) = h$. Therefore, both $hh'$ and $h'h$ are images of $I_t$; hence, the edge is undirected.

We make an observation about the indicator construction for edge-coloured graphs which is unique to edge-coloured graphs. The edge-coloured graph $H^*$ has has multiplicity $m$ – one edge-colour for each indicator. However, multiplicity of $H^*$ is in no way related to the multiplicity of $H$. The edge-coloured graph $H^*$ can have fewer, the same, or more edge-colours than $H$. Nevertheless, each indicator, $I_t$, must have the same multiplicity as $H$.

**Lemma 2.1.1** *Let $H^*$ be defined as above. Then $H^*$-COL polynomially transforms to $H$-COL.*

**Proof.** Given an edge-coloured graph $G$, let ${}^*G$ be the edge-coloured graph obtained by replacing each edge of colour $t$, say $uv$, with the edge-coloured graph $I_t$, identifying $u$ with $i_t$ and $v$ with $j_t$. Note that $G$ has the same multiplicity as $H^*$ (the number of indicators) and ${}^*G$ has the same multiplicity as $H$ (the common multiplicity of all the indicators). It is straightforward from the definitions that ${}^*G \to H$ if and only if $G \to H^*$. This was done, in [19], for the case of graphs.■

See Figure 2.1 for an example of the indicator construction.

It is possible that $i_t$ and $j_t$ may map to the same vertex in $H$. This will produce a loop of colour $t$ in $H^*$. One must be careful to avoid constructing loops in the case of

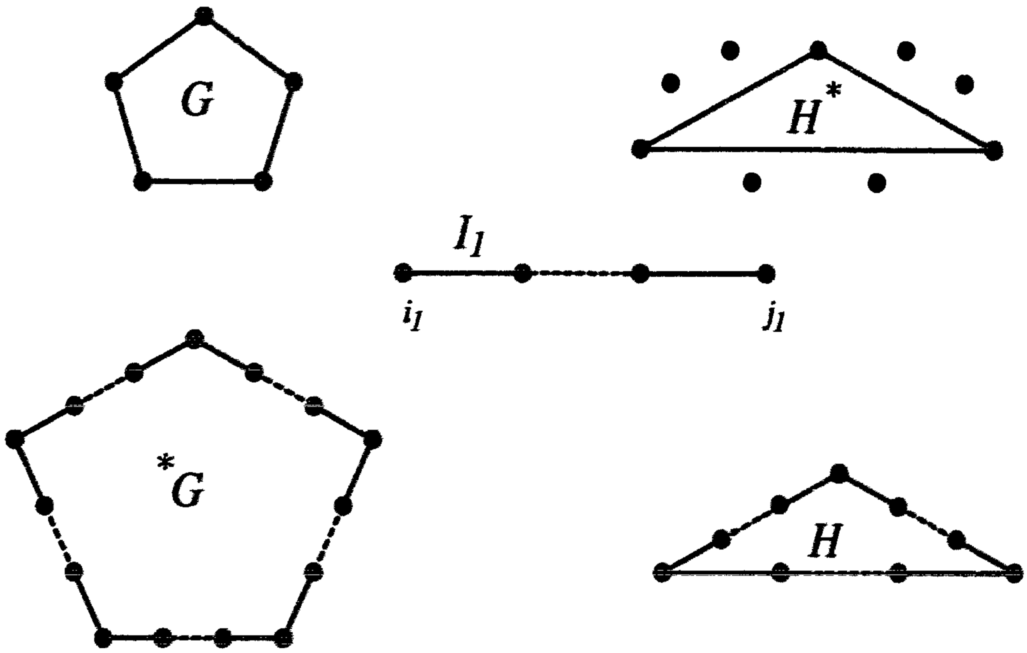


Figure 2.1: An example of the indicator construction

graphs or directed graphs, since a loop in $H^*$ makes $H^*$-COL trivial. This may not be the case when working with edge-coloured graphs. It is possible to have loops in $H^*$ and still $H^*$-COL may not be trivial.

Consider the example in Figure 2.2. Let $H$ be the edge-coloured graph with vertex set $\{0,1,2\}$; blue edge-set $\{01,02,11,12\}$; and red edge-set $\{00,12\}$. Let $I_1$ be a single blue edge with end-points $i_1$ and $j_1$. Let $I_2$ be a path of length three consisting of a red edge, a blue edge, and a red edge. Let $i_2$ and $j_2$ be the end-points of the path. The result of the indicator construction with respect to $((I_1,i_1,j_1),(I_2,i_2,j_2))$ is the edge-coloured graph $H^*$ shown in Figure 2.2. Despite the fact that $H^*$ contains loops, $H^*$-COL is NP-complete. One can see this through a second application of the indicator construction. Let $I_1^*$ be a red-blue digon. That is, $I_1^*$ is an edge-coloured graph on two vertices, $i_1^*$ and $j_1^*$, where $i_1^*j_1^*$ is a red edge and a blue edge. The result

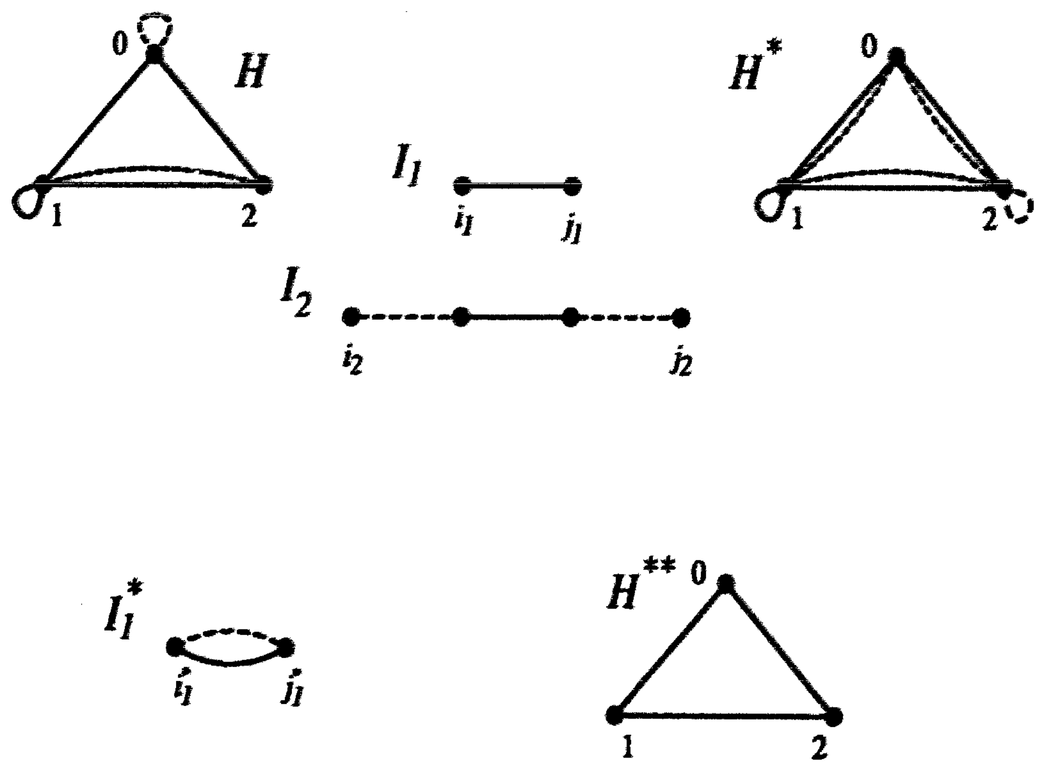


Figure 2.2: A hard $H^*$-COL problem with loops

of the indicator construction with respect to $(I_1^*, i_1^*, j_1^*)$, applied to $H^*$, is $H^{**}$. The edge-coloured graph $H^{**}$ is simply a $K_3$ on one edge-colour. It is well known that $H^{**}$-COL is NP-complete, see [12]. Notice, the second application of the indicator construction is one where the result has fewer edge-colours than does the original edge-coloured graph.

### 2.1.2 The Sub-indicator Construction

Let $J$ be a fixed edge-coloured graph with specified vertices $j, k_1, k_2, \ldots, k_t$. The sub-indicator construction, with respect to $J, j, k_1, \ldots, k_t$, transforms a given core $H$ with specified vertices $h_1, \ldots, h_t$, to an induced subgraph $H^-$ of $H$. The subgraph $H^-$, on vertex set $V^-$, is defined as follows. Let $W$ be the edge-coloured graph obtained by taking disjoint copies of $H$ and $J$ and identifying vertices $h_i$ and $k_i$ (for $i = 1, 2, \ldots, t$). A vertex, $h$, of $H$ belongs to $V^-$ if and only if there is a retraction of $W$ to $H$ mapping $j$ to $h$. An example of the subindicator construction is given in Figure 2.3.

**Lemma 2.1.2** *Let $H$ be a core. Then $H^-$-COL polynomially transforms to $H$-COL.*

**Proof.** Given an edge-coloured graph $G$ we construct an edge-coloured graph ${}^-G$ by taking disjoint copies of $G, H$, and $|V(G)|$ copies of $J$. Identify $h_i$ with $k_i$ in each copy of $J$ for $i = 1, 2, \ldots, t$. For each vertex $g$ in $V(G)$, identify $g$ with $j$ in the $g^{th}$ copy of $J$. If there is a homomorphism of ${}^-G \rightarrow H$, then the copy of $H$ in ${}^-G$ must map *onto* $H$ since $H$ is a core. It is now easy to see that ${}^-G \rightarrow H$ if and only if $G \rightarrow H^-$. ∎

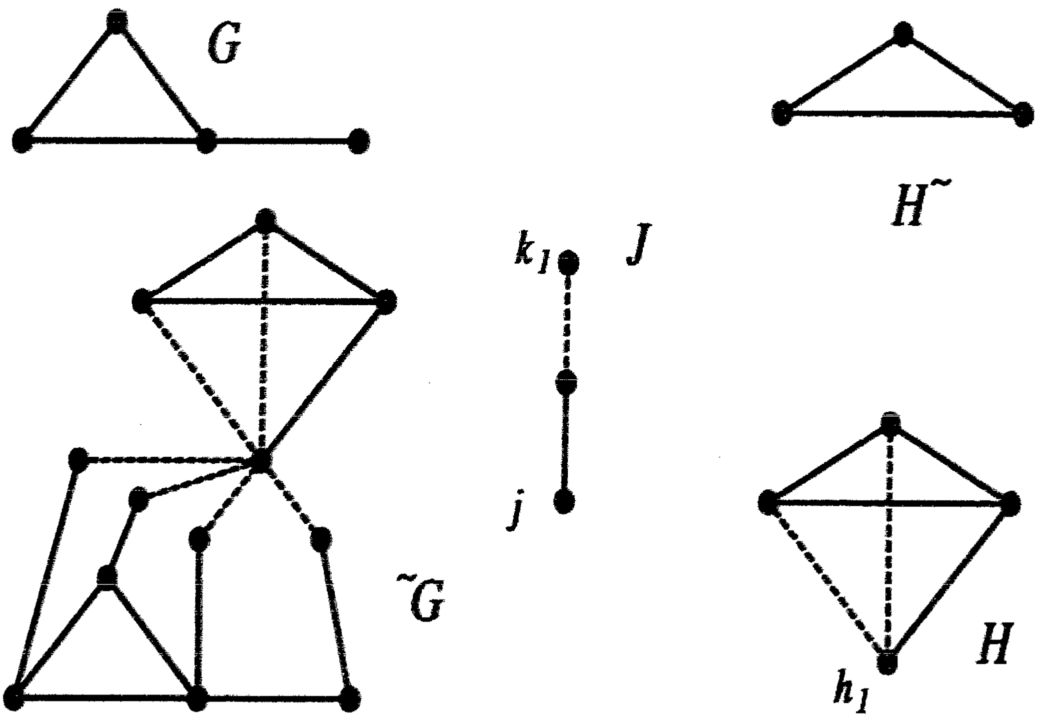


Figure 2.3: An example of the subindicator construction

## 2.2 The Forcing Lemma

Many times in life the easiest decisions are the ones made for us. The same is true when solving $H$-colouring problems. Consider an edge-coloured graph, $H$, where each vertex is incident with at most one edge of each colour. Let $G$ be a connected, edge-coloured graph. As usual, we are interested in the existence or nonexistence of a homomorphism $f : G \to H$. Let $g$ be a vertex in $G$ and $h$ be a vertex in $H$. In this section we observe that if $G$ admits a homomorphism, $f$, to $H$ such that $f(g) = h$, then $f$ is unique.

**Lemma 2.2.1** *Let $H$ be an edge-coloured graph such that each vertex of $H$ is incident with at most one edge of each colour. Given a connected, edge-coloured graph $G$, $g \in V(G)$ and $h \in V(H)$, if there exists a homomorphism $f : G \to H$, such that*

*$f(g) = h$, then $f$ is unique.*

**Proof.** We prove the result by induction on the number of edges of $G$. Let $H$, $G$, $g$, and $h$ be as above. Suppose $|E(G)| = 0$. Then $G$ consists of a single vertex $G$ and the homomorphism $f(g) = h$ is unique. Therefore, suppose the lemma holds for all $G$, with $|E(G)| \leq m$. Let $G$ be a connected, edge-coloured graph with $m+1$ edges. Suppose there exist two homomorphisms $f_1 : G \to H$ and $f_2 : G \to H$ such that $f_1(g) = f_2(g) = h$. We show $f_1 = f_2$. Since $G$ is connected, we know there exists $g' \in N(g)$. Suppose $gg' \in E_j(G)$. Since $G \to H$, there exists $h' \in N_j(h)$; moreover, by our assumptions on $H$, $h'$ is unique and hence $f_1(g') = f_2(g') = h'$.

Consider the edge-coloured graph $G\backslash\{gg'\}$. If $G\backslash\{gg'\}$ is connected, then by induction $f_1 = f_2$. On the other hand, if $G\backslash\{gg'\}$ is disconnected, then it consists of two components $G_1$ and $G_2$, such that $g \in G_1$ and $g' \in G_2$. Recall $f_1(g) = f_2(g) = h$. By induction $f_1 = f_2$ on $G_1$. Similarly, $f_1(g') = f_2(g') = h'$ and hence $f_1 = f_2$ on $G_2$. Since $V(G) = V(G_1) \cup V(G_2)$, we have $f_1 = f_2$ on $G$. ∎

We have the following immediate corollary. In the following we use the standard notation $Hom(G, H) = \{f \mid f : G \to H \text{ such that } f \text{ is a homomorphism}\}$.

**Corollary 2.2.2** *Let $H$ be an edge-coloured graph such that each vertex of $H$ is incident with at most one edge of each colour. Suppose $G$ is a connected edge-coloured graph. Then $|Hom(G, H)| \leq |V(H)|$.*

**Proof.** Choose a fixed $g \in V(G)$. For each $h \in V(H)$ there is at most one $f \in Hom(G, H)$ such that $f(g) = h$. ∎

We now prove a result which will be used repeatedly throughout this thesis. Namely, the set $Hom(G, H)$ can be constructed in $O(|E(G)|)$ time.

**Lemma 2.2.3 (The Forcing Lemma)** *Let $H$ be an edge-coloured graph such that each vertex of $H$ is incident with at most one edge of each colour and let $G$ be a connected edge-coloured graph. Then the elements of $Hom(G, H)$ can be generated in $O(|E(G)|)$ time.*

**Proof.** Let $V(H) = \{h_1, h_2, \ldots, h_n\}$. Let $G$ be a connected edge-coloured graph.

Suppose $S$ is a subset of $V(G)$ and $f$ is a homomorphism from the subgraph induced by $S$ to $H$. Let $u$ be a vertex in $V(G)\backslash S$. We say $f$ *is extendable to* $u$ if there exists $h \in V(H)$ such that by defining $f(u) = h$ one obtains that $f$ is a homomorphism from the subgraph induced by $S \cup \{u\}$ to $H$.

Since $H$ has the property that each vertex is incident with at most one edge of each colour, the following observation is true. Suppose $S$ is a subset of $V(G)$, $u \in V(G)\backslash S$, and $u$ is adjacent to a vertex $v \in S$. Then, if $f$ is extendable to $u$, this extension is unique. Suppose $u$ and $v$ are joined by an edge of colour $t$. The set $N_t(f(v))$ contains at most one element and this element, if it exists, must be the image of $u$ under $f$.

We now describe the algorithm. Choose a vertex $g \in V(G)$. Repeat the following steps for each $h_i \in V(H)$.

- Set $f(g) = h_i$ and $S = \{g\}$.
- While $V(G)\backslash S$ is not empty:
  - Choose $u$ in $V(G)\backslash S$ such that $u$ is adjacent to some $v \in S$.

  - If $f$ can be extended to $u$, then extend $f$ else STOP.
  - Add $u$ to $S$.
- Add $f$ to $Hom(G, H)$.

We now prove the algorithm correctly generates $Hom(G, H)$. Suppose some function $f$ is added to $Hom(G, H)$ by the algorithm. This implies $f$ was successfully extended to all of $V(G)$ which implies $f$ is a homomorphism from $G$ to $H$. On the other hand, suppose $f : G \to H$ is a homomorphism. There is an $i$ such that $f(g) = h_i$. On the $i^{th}$ iteration of the algorithm, $f(g) = h_i$. The algorithm then attempts to extend $f$ to $V(G)$. Since $f$ is a homomorphism this extension is possible and unique; hence, $f$ is added to $Hom(G, H)$.

Finally, we show the algorithm runs in $O(|E(G)|)$ time. The size of $H$ is fixed, so we only need to show that the time to extend one $f$ is $O(|E(G)|)$. If the homomorphism $f$ can be extended to $u$, then this image of $u$ under $f$ is unique. To test if $f$ is a homomorphism on $S \cup \{u\}$, we check that for each $v$ in $N(u) \cap S$ such that $uv \in E_j(G)$ implies $f(u)f(v) \in E_j(H)$. This requires $O(deg(u))$ time. Hence the entire algorithm requires $O(|E(G)|)$ time. ∎

A final point to consider is what happens when $G$ is not connected? If one is solely interested in the question "Does $G \to H$?", then the Forcing Lemma can be used on each component of $G$. The edge-coloured graph $G$ is a YES instance of $H$-COL if and only if each component is a YES instance of $H$-COL. However, if one is interested in constructing $Hom(G, H)$, then a disconnected edge-coloured graph $G$ can cause problems. Let $G_1, G_2, \ldots, G_m$ be the components of $G$. From the above results we see there are at most $n$ homomorphisms from $G_i$ to $H$ where $n = |V(H)$. To construct a homomorphism from $G$ to $H$, we need to choose a homomorphism from each component to $H$. Therefore, there could be as many as $n^m$ homomorphisms from $G$ to $H$. Hence, one must be careful when using the Forcing Lemma to generate $Hom(G, H)$, since this set can have exponential size if $G$ is disconnected.

The Forcing Lemma is a result that naturally extends to directed, edge-coloured graphs. If the condition on $H$ is changed to "Each vertex is incident with at most one in-arc and at most one out-arc of each colour", then the result still holds.

A final observation we make is the obvious interpretation of this result in the context of multiplicity one. The only connected graphs with degree at most one are $K_1$ and $K_2$. Digraphs that have indegree and outdegree at most one at each vertex are directed paths and directed cycles. These graphs $H$ yield polynomial $H$-colouring problems, as is well known [12].

## 2.3 Reduction to 2-Satisfiability

The use of propositional logic problems is quite common in complexity theory, especially for proving NP-completeness. In this section we describe a method for constructing polynomial time algorithms using 2-Satisfiability (2SAT). Formally we define 2SAT as:

**2-Satisfiability (2SAT)**

INSTANCE: A set $\mathcal{U}$ of boolean variables, a collection $\mathcal{C}$ of clauses over $\mathcal{U}$ such that each $C \in \mathcal{C}$ has at most two literals.

QUESTION: Is there a satisfying truth assignment for $\mathcal{C}$?

In this definition a clause is a disjunction of variables and a satisfying truth assignment is assignment of true or false to each variable such that each clause is true. This problem is solvable in polynomial time. See [12], [9], and [29]. We describe a method for polynomially transforming $H$-COL to 2SAT.

Suppose we are given an edge-coloured graph $H$. We assign to each vertex in $H$ a bit string of length $n$, i.e., a string of $n$ 1's and 0's. Given an instance of $H$-COL, say $G$, we construct $n$ boolean variables for each vertex in $G$. In addition, we construct clauses on this set of variables in such a way that the clauses have a satisfying truth assignment if and only if $G \to H$. For example, let $u$ be a vertex in $G$. We denote the $n$ variables corresponding to $u$ as $u_n, u_{n-1}, \ldots, u_1$. A truth assignment on these variables can be represented as a bit string of length $n$. That is, $u_i$ is true if and only if position $i$ of the bit string is a one. Since we have assigned each vertex in $H$ a bit string of length $n$, there is a natural correspondence between a truth assignment for $u_n, u_{n-1}, \ldots, u_1$ and an image for vertex $u$ in $H$.

This is an idea whose simplicity can easily be shrouded by definitions and lemmas. Let us consider an example. Let $H$ be an edge-coloured graph on three vertices $\{x, y, z\}$ and two edge-colours. The blue edges are $\{xy, xz\}$ and the red edges are $\{xy, yz\}$. To each vertex in $H$, we assign a bit string of length two:

$$\begin{aligned} x &\to 10 \\ y &\to 01 \\ z &\to 00 \end{aligned}$$

Let $G$ be an instance of $H$-COL. In particular, let $G$ be the edge-coloured graph on three vertices, $\{u, v, w\}$, with blue edge set $\{uv\}$ and red edge set $\{vw\}$. We construct

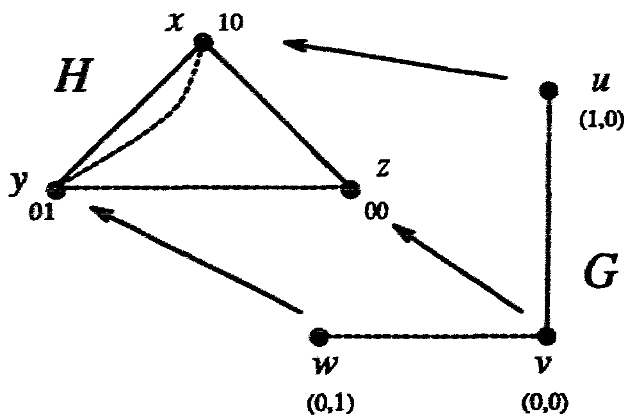


Figure 2.4: Example of 2SAT reduction

the corresponding instance of 2SAT. The set of variables is $\mathcal{U} = \{u_2, u_1, v_2, v_1, w_2, w_1\}$ and the set of clauses is $\mathcal{C}$ described below. The bit string 11 has not been used in labeling $H$; therefore, in any truth assignment we need to avoid assigning $(1,1)$ to $(u_2, u_1)$, $(v_2, v_1)$, or $(w_2, w_1)$. This can be accomplished by placing the clauses $(\neg u_2 \vee \neg u_1)$, $(\neg v_2 \vee \neg v_1)$, and $(\neg w_2 \vee \neg w_1)$in $\mathcal{C}$. Secondly, we construct a set of clauses for each edge in $G$ which will assure that the edge is preserved:

| *edge* | *clauses* |
|---|---|
| $uv$ | $(u_2 \vee v_2) \wedge (\neg u_2 \vee \neg v_2)$ |
| $vw$ | $(v_1 \vee w_1) \wedge (\neg v_1 \vee \neg w_1)$ |

To see how these clauses were chosen, examine the red edges in $H$. The bit string labels in $H$ have the property that $ab$ is a red edge in $H$ if and only if the first bit (reading right to left) of the label of $a$ is different from the first bit of the label of $b$. Hence, given a red edge $gg' \in E_{red}(G)$, a truth assignment of $(g_2, g_1)$ and $(g'_2, g'_1)$ corresponds to an edge preserving mapping of $gg'$ to $H$ if and only if $g_1 \neq g'_1$. The clause for red edges above (namely for $vw$) is an exclusive or of $v_1$ and $w_1$. That is, it has a truth assignment if and only if $v_1 \neq w_1$. This is precisely the condition we require for a homomorphism. The same argument can be applied to the blue edges using the second bit of the labels.

The following truth assignment satisfies all the clauses in $C$:

$$\begin{aligned} (u_2, u_1) &= (1,0) \\ (v_2, v_1) &= (0,0) \\ (w_2, w_1) &= (0,1) \end{aligned}$$

Since $(u_2, u_1) = (1,0)$, we map vertex $u \in G$ to $x \in H$. Similarly, we map $v$ to $z$ and $w$ to $y$. It is easy to check that this is a homomorphism from $G$ to $H$.

In the above example there are two steps in our reduction. First, we put the clauses $(\neg a_2 \vee \neg a_1)$ in $C$ for all $a \in V(G)$. This insures that the truth assignment $(a_2, a_1) = (1,1)$ is never used, since it does not satisfy $(\neg a_2 \vee \neg a_1)$. This must be done since 11 is not a label in $H$. Secondly, we construct clauses for each edge, $ab$ in $G$, such that a truth assignment for $(a_2, a_1)$ and $(b_2, b_1)$ exists if and only if the mapping induced by this truth assignment maps $ab$ onto an edge of the same colour. Therefore, the constructed instance of 2SAT contains: firstly, clauses that describe the valid labels in $H$; and secondly, clauses that describe the valid mapping of edges of each colour.

Formally, we call a set, $S$, of bit strings of length $n$ *2SAT-describable* if there is an instance of 2SAT over the variables $\{s_n, s_{n-1}, \ldots, s_1\}$ such that $t : \{s_n, \ldots, s_1\} \rightarrow \{0,1\}$ is a satisfying truth assignment if and only if $t(s_n)\ldots t(s_1) \in S$. Hence, the first step in our reduction is to label the vertices of $H$ with a 2SAT describable set.

Since any truth assignment satisfies an empty set of clauses we have the following observation.

**Observation 2.3.1** *Let $S$ be the set of all $2^n$ bit strings of length $n$. Then $S$ is 2SAT-describable.*

There are exactly four bit strings of length two. In our example above, we used three of the four to label $H$. In fact, any three of the of the four bit strings form a 2SAT-describable set.

**Lemma 2.3.2** *Let $S$ be any set of three bit strings of length two. Then $S$ is 2SAT describable.*

**Proof.** There are four bit strings of length two and by assumption $S$ consists of three of them. Let $xy$ be the one bit string of length two not in $S$. The instance of 2SAT over $\mathcal{U} = \{u, v\}$ with the following single clause has a truth assignment if and only if $(u, v) \neq (x, y)$. Notice $(u, v)$ satisfies $\neg(x \wedge y)$ if and only if $(u, v)$ satisfies $(\neg x \vee \neg y)$. Hence we have the following four cases:

| *Value of $xy$* | *Set of Clauses* |
|---|---|
| 00 | $\mathcal{C} = \{(u \vee v)\}$ |
| 01 | $\mathcal{C} = \{(u \vee \neg v)\}$ |
| 10 | $\mathcal{C} = \{(\neg u \vee v)\}$ |
| 11 | $\mathcal{C} = \{(\neg u \vee \neg v)\}$ |

A truth assignment for $(u, v)$ satisfies $\mathcal{C}$ if and only if $(u, v) \neq (x, y)$. That is, if and only if $(u, v)$ is a member of $S$. ∎

The following theorem gives sufficient conditions to allow an $H$-COL problem to be polynomially reduced to a 2SAT problem. Since 2SAT is polynomial, such an H-COL problem is polynomial. There are many polynomial $H$-COL problems that are not of the form described below; hence, the conditions are not necessary for a polynomial reduction to 2SAT. Note that every polynomial problem trivially polynomially reduces to 2SAT.

We now describe a class of graphs that can easily be reduced to 2SAT. Consider a graph (one edge-colour) where the vertices are partitioned into two sets, $X$ and $Y$. We call the graph *2SAT amiable with respect to* $(X, Y)$ if and only if the following three conditions hold:

(i) either all edges between $X$ and $Y$ are present or no edges between $X$ and $Y$ are present;

(ii) either $X$ induces a clique with loops or $X$ induces an independent set;

(iii) either $Y$ induces a clique with loops or $Y$ induces an independent set.

**Theorem 2.3.3** *Let $H$ be an edge-coloured graph with multiplicity $k$. Suppose the vertices of $H$ have been labelled with a 2SAT-describable set $S$ using bit strings of length $k$. For all $t \leq k$, let $X_t = \{v \in V(H) :$ the label of $v$ has a 0 in position $t$ $\}$. Let $Y_t = V(H) \backslash X_t$. Suppose $(V(H), E_t(H))$ is 2SAT amiable with respect to $(X_t, Y_t)$. Then $H$-COL can be polynomially transformed to 2SAT.*

**Proof.** Let $G$ be an instance of $H$-COL. Consider any edge colour $i$. We describe the clauses to be added to $C$ for each $uv \in E_i(G)$. By assumption $(V(G), E_i(G))$ is 2SAT amiable with respect to $(X_i, Y_i)$; therefore, one of the eight cases in Figure 2.5

| | *Edges between* $X_i$ *and* $Y_i$ | *Graph induced by* $X_i$ | *Graph induced by* $Y_i$ | *Add clause to* $C$ |
|---|---|---|---|---|
| 1 | None | Independent | Independent | $(\neg u_i) \wedge (u_i)$ |
| 2 | None | Clique with loops | Independent | $(\neg u_i) \wedge (\neg v_i)$ |
| 3 | None | Independent | Clique with loops | $(u_i) \wedge (v_i)$ |
| 4 | None | Clique with loops | Clique with loops | $(u_i \vee \neg v_i) \wedge (\neg u_i \vee v_i)$ |
| 5 | All | Independent | Independent | $(u_i \vee v_i) \wedge (\neg u_i \vee \neg v_i)$ |
| 6 | All | Clique with loops | Independent | $(\neg u_i \vee \neg v_i)$ |
| 7 | All | Independent | Clique with loops | $(u_i \vee v_i)$ |
| 8 | All | Clique with loops | Clique with loops | $(u_i \vee \neg u_i)$ |

Figure 2.5: Clauses for 2SAT reduction

must hold. Add the appropriate clause from Figure 2.5 to $C$ for each edge colour $i$ and for all edges $uv \in E_i(G)$.

We now confirm that these clauses are correct. Consider an arbitrary edge colour $i$ calling it blue. Let $uv$ be a blue edge. If case one holds, then there are no blue edges in $H$. The clause $(\neg u_i) \wedge (u_i)$ has no satisfying truth assignment. Therefore, $uv$ maps to $H$ if and only if the clause is satisfied. If case 2 holds, then all blue edges in $H$ have both ends in $X_i$. That is, $uv$ maps to $H$ if and only if both $u_i = 0$ and $v_i = 0$. The clause $(\neg u_i) \wedge (\neg v_i)$ is satisfied if and only if $(u_i, v_i) = (0, 0)$. If case 3 holds, then all blue edges in $H$ have both ends in $Y_i$. That is, $uv$ maps to $H$ if and only if both $u_i = 1$ and $v_i = 1$. The clause $(u_i) \wedge (v_i)$ is satisfied if and only if $(u_i, v_i) = (1, 1)$. If case 4 holds, then all blue edges either have both ends in $X_i$ or both ends in $Y_i$. The clause $(u_i \vee \neg v_i) \wedge (\neg u_i \vee v_i)$ is satisfied if and only if $u_i = v_i$. That is, if and only if $(u_i, v_i) = (0, 0)$ or $(u_i, v_i) = (1, 1)$.

If case 5 holds, then all blue edges in $H$ have exactly one end in $X_i$ and exactly one end in $Y_i$. The clause $(u_i \vee v_i) \wedge (\neg u_i \vee \neg v_i)$ is satisfied if and only if $(u_i, v_i) = (1, 0)$ or $(u_i, v_i) = (0, 1)$. In other words, $uv$ maps to $H$ if and only if the clause is satisfied.

If case 6 holds, then all blue edges in $H$ have at least one edge in $X_i$. The clause $(\neg u_i \vee \neg v_i)$ is satisfied if and only if at least one of $\{u_i, v_i\}$ is 0. If case 7 holds, then all blue edges in $H$ have at least one edge in $Y_i$. The clause $(u_i \vee v_i)$ is satisfied if and only if at least one of $\{u_i, v_i\}$ is true. Finally, in case 8, all possible blue edges in $H$ are present. Therefore, $uv$ can map to any pair of vertices in $H$. The clause $(u_i \vee \neg u_i)$ is satisfied by any truth assignment to $(u_i, v_i)$.

With these clauses inserted into $C$ for each edge colour and all appropriate variables, the instance of 2SAT has a satisfying truth assignment if and only if $G \to H$. ∎

## 2.4 Divide and Conquer

In this section we describe two techniques for studying the complexity of $H$-COL based on the complexity of $H'$-COL where $H'$ is a subgraph of $H$. We also use a third technique, the "bipartite decomposition lemma", similar to these two, which requires results from Chapter three. Hence, we present it at the end of Chapter three.

### 2.4.1 The Join Lemma

We begin by studying the case when $H$ is the join of two smaller edge-coloured graphs.

**Lemma 2.4.1** *Let $H_1$ and $H_2$ be two edge-coloured graphs with multiplicity $k$ such that $H_1 \to H_2$. Let $H$ be the join of $H_1$ and $H_2$ with respect to $\{1, 2, \ldots, k, k+1\}$ and $H'$ be the join of $H_1$ and $H_1$ with respect to $\{1, 2, \ldots, k\}$. Then $H'$-COL polynomially transforms to $H$-COL.*

**Proof.** Let $G$ be an instance of $H'$-COL. Let colour $k+1$ be blue. Construct a graph $X$ by taking two copies of $G$, say $G^A$ and $G^B$, and joining corresponding vertices with blue edges so that a blue matching between $G^A$ and $G^B$ results. Note we can assume $G$ itself has no blue edges, since $H'$ has no blue edges. A blue edge in $G$ implies $G$ is trivially a NO instance of $H'$-COL. We show $X \to H$ if and only if $G \to H'$.

Suppose $h : G \to H'$ is a homomorphism. Let the two copies of $H_1$ in $H'$ be denoted $H_1^A$ and $H_1^B$. Let $S$ be the induced subgraph on the set of vertices in $G$ that is mapped to $H_1^A$ by $h$. Similarly let $\overline{S}$ be the induced subgraph on the set of vertices in $G$ that is mapped to $H_1^B$ by $h$. It is clear that $V(S)$ and $V(\overline{S})$ partition the vertices of $G$ into two sets. Moreover, $S \cup \overline{S}$ admits a homomorphism to $H_1$. Since $H_1 \to H_2$, it is also the case that $S \cup \overline{S} \to H_2$. It is now easy to see that $X \to H$.

On the other hand, suppose $f : X \to H$ is a homomorphism. As above we can partition the vertices of $X$ into two sets, those that map to $H_1$ and those that map to $H_2$. Let $S^A$ be the induced subgraph of $G^A$ on the set of vertices of $G^A$ that is mapped to $H_1$ by $f$. Let $\overline{S^B}$ be the induced subgraph of $G^B$ on the set of vertices of $G^B$ that is mapped to $H_1$ by $f$. Let $\overline{S^A}$ be the induced subgraph of $G^A$ on the set of vertices of $G^A$ that is mapped to $H_2$ by $f$. By the construction of the matching in $X$, $\overline{S^A}$ is isomorphic to $\overline{S^B}$. Since $\overline{S^B}$ admits a homomorphism to $H_1$, it is the case that $\overline{S^A}$ admits a homomorphism to $H_1$. It is now easy to construct a homomorphism $G \to H'$. ∎

In the proof above, the edge-coloured graph constructed from the two copies of $G$, contains a perfect matching in blue. We have the following immediate corollary.

**Corollary 2.4.2** *Let $H$ and $H'$ be as above. If $H'$-colouring is NP-complete, then $H$-colouring is NP-complete even when the input is restricted to edge coloured graphs* *which contain a perfect matching in blue.*

### 2.4.2 The Dominating Loop Lemma

We now consider a method for reducing the target edge-coloured graph $H$ by a single vertex. In the following lemma we are interested in finding a vertex in $H$ incident with only one colour and adjacent to all vertices including itself.

**Lemma 2.4.3 (Dominating Loop Lemma)** *Let $H$ be an edge-coloured graph with a vertex $v$ such that $v$ is incident with edges of only colour $i$. Further suppose that $v$ is adjacent (in colour $i$) to all vertices in $H$ including itself. Let $G$ be an edge-coloured graph and let $X$ be the set of all vertices in $G$ incident with only colour $i$. Then $G \to H$ if and only if $G \backslash X \to H \backslash \{v\}$.*

**Proof.** Suppose $H$ satisfies the condition of the lemma and vertex $v$ is blue only. Firstly, suppose $G \to H$. The vertices that map to $v$ must be a subset of $X$ since $v$ is blue only. Therefore $G \backslash X$ must map to $H \backslash \{v\}$. On the other hand, suppose $f : G \backslash X \to H \backslash \{v\}$. Define a new homomorphism $g$ where

$$g(u) = \begin{cases} f(u) & \text{if } u \notin X \\ v & \text{if } u \in X \end{cases}$$

Since $v$ is adjacent in blue to all other vertices of $H$ and $X$ is adjacent only in blue to $G \backslash X$, it is easy to verify that $g$ is a homomorphism. ∎

**Corollary 2.4.4** *Let $H$ and $v$ be as above. Then $H$-COL $\propto$ $H \backslash \{v\}$-COL.*

The above corollary says if $H$-COL is NP-complete, then $H \backslash \{v\}$-COL is NP-complete. If $H \backslash \{v\}$-COL is polynomial, then $H$-COL is polynomial. We also present a partial converse to Corollary 2.4.4.

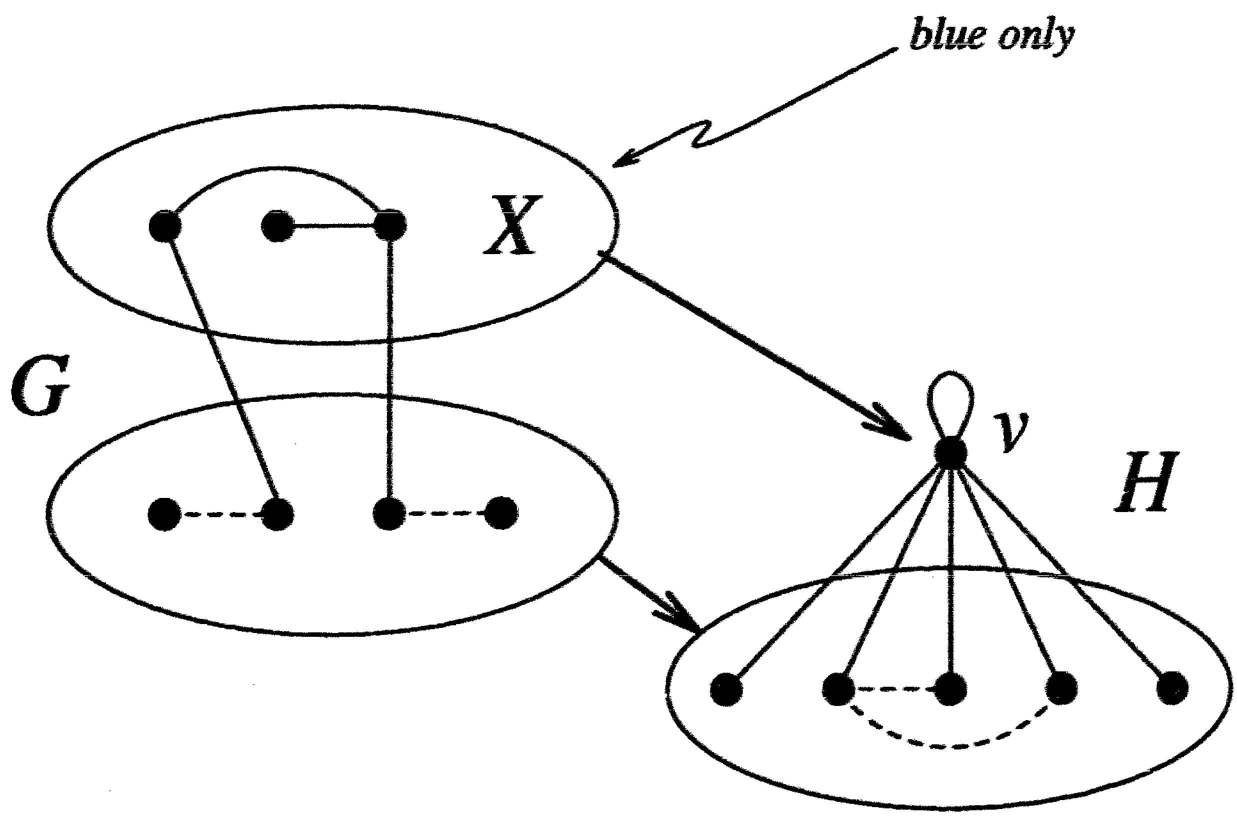


Figure 2.6: Example of $H$ with a dominating loop.

**Corollary 2.4.5** *Let $H$ be a two-edge-coloured graph that is a core. Suppose $H$ contains a vertex $v$ that is incident only with edges of one colour. Further suppose that $v$ is adjacent (in that colour) to every vertex in $H$ including itself. Then $H$-COL and $H\backslash\{v\}$-COL are polynomially equivalent.*

**Proof** We know from Corollary 2.4.4 that $H$-COL $\alpha$ $H\backslash\{v\}$-COL. We now prove that $H\backslash\{v\}$-COL $\alpha$ $H$-COL. Suppose without loss of generality the edges of $H$ are blue and red and the vertex $v$ is incident with only blue edges. Since $H$ is a core, it can not have any vertices other than $v$ that are incident with only blue edges. That is, every vertex in $H\backslash\{v\}$ is incident with a red edge. Let $G$ be an instance of $H\backslash\{v\}$-COL. For each vertex $u \in V(G)$ add a new vertex $u'$ and join $u$ to $u'$ with a red edge. Call this new graph $G'$. Since all vertices in $H\backslash\{v\}$ are incident with red edges $G \to H\backslash\{v\}$ if and only if $G' \to H\backslash\{v\}$. Since all vertices in $G'$ are incident with a red edge, $G' \to H\backslash\{v\}$ if and only if $G' \to H$. Therefore, $G \to H\backslash\{v\}$ if and only if $G' \to H$. The result follows. ∎

# Chapter 3

# The Homomorphism Factoring Problem

## 3.1 General Results

The results in this chapter come in two flavours. Some results hold for the most general systems considered in this thesis – edge-coloured, directed graphs. Other results are written in terms of graphs and digraphs (i.e. one edge-colour). This situation is acceptable in the sense that the general results are those that show the existence of polynomial time algorithms and the specific (graph) results prove NP-completeness of certain problems. That is, we show certain problems are easy even in the most general case; whereas, we show other problems are hard even in a restricted case.

For graphs, we know from [19] that to test for the existence of a homomorphism to any fixed nonbipartite graph is NP-complete. An interesting question is to find the complexity of $H$-colouring when the input is restricted to a particular set of graphs. For example, any graph that admits a homomorphism to $C_5$ must also admit a homomorphism to $K_3$ since $C_5 \to K_3$. Furedi, Griggs, and Kleitman [11] asked if knowing $G$ is 3-colourable makes testing $C_5$-colourability any easier. In [6] we prove that it is not easier, even when a homomorphism to $K_3$ is provided. Specifically, we consider in [6] the following restricted homomorphism problem RHP. We state the problem here in the more general context of edge-coloured graphs, although in [6] we only consider graphs. Thus let $H$ and $Y$ be fixed edge-coloured graphs.

**Restricted Homomorphism Problem** $RHP(H, Y)$

INSTANCE: An edge-coloured graph $G$ and a homomorphism $g : G \to Y$.

QUESTION: Does there exist a homomorphism $f : G \to H$.

For the case of undirected, uncoloured graphs, we proved in [6] the following result:

**Theorem 3.1.1** *Let $H$ be a loopless graph. If $\omega(H) < k \leq \chi(H)$, then $RHP(H, K_k)$ is NP-complete. Otherwise, $k \leq \omega(H) \leq \chi(H)$ and $RHP(H, K_k)$ is polynomial.*

We now consider a problem, due to Sabidussi and Tardiff, which is closely related to, and perhaps more natural than, the restricted homomorphism problem. We begin by stating this problem in the context of edge-coloured digraphs since the first set of results holds for this general case. We will later restrict our attention to graphs. Let $H$ and $Y$ be two edge-coloured digraphs and $h : H \to Y$ a homomorphism.

**Homomorphism Factoring Problem** $HFP(H, h, Y)$

INSTANCE: An edge-coloured digraph $G$ and a homomorphism $g : G \to Y$.

QUESTION: Does there exist a homomorphism $f : G \to H$ such that $h \circ f = g$?

We begin by defining an indicator construction for use with the Homomorphism Factoring Problem. The indicator construction in Chapter Two is stated in terms of $m$ indicators, $I_1, I_2, \ldots, I_m$. Consequently, the result of the indicator construction has $m$ edge-colours. Our discussion here is in terms of one indicator for simplicity and in view of the fact that the indicator construction is used in the one edge-colour context in this chapter. However, the HFP indicator construction and the following lemma have a natural generalization to $m$ indicators.

Let $H$ and $Y$ be two edge-coloured digraphs and let $h : H \to Y$ be a homomorphism. Let $I$ be an edge-coloured digraph with distinguished vertices $i$ and $j$. Further suppose $t : I \to Y$ is a homomorphism. The indicator construction, with respect to $(I, i, j, t)$, transforms $H$ and $Y$ into two new edge-coloured digraphs $H^\#$ and $Y^\#$. The vertex-set of $H^\#$ is $V(H)$. Given two vertices $u$ and $v$ in $V(H)$, $uv \in E(H^\#)$ if and only if there is a homomorphism $r : I \to H$ such that $r(i) = u$, $r(j) = v$, and $h \circ r = t$. The vertex-set of $Y^\#$ is $V(Y)$ and the edge-set is the single arc $(t(i), t(j))$.

Consider the example in Figure 3.1. The graph $H$ is $C_9$ and the graph $Y$ is $C_3$. The numbers beside the vertices in $H$ define the homomorphism $h : H \to Y$. All vertices with 0 beside them are mapped to the vertex labelled 0 in $Y$. Similarly, the graph $I$ is a $P_3$ and the homomorphism $t : I \to Y$ is also marked in the figure. The pair $(u, v)$ is an arc in $H^\#$ if and only if $I$ admits a homomorphism to $H$ with $i$ mapping to $u$ of colour 0 and $j$ mapping to $v$ of colour 0. The edge-coloured digraph $Y^\#$ contains the single arc from $t(i) = 0$ to $t(j) = 0$.

Part of the description of $HFP(H, h, Y)$ is the homomorphism $h$. It is important to note the homomorphism, $h : H \to Y$, is also a homomorphism from $H^\#$ to $Y^\#$.

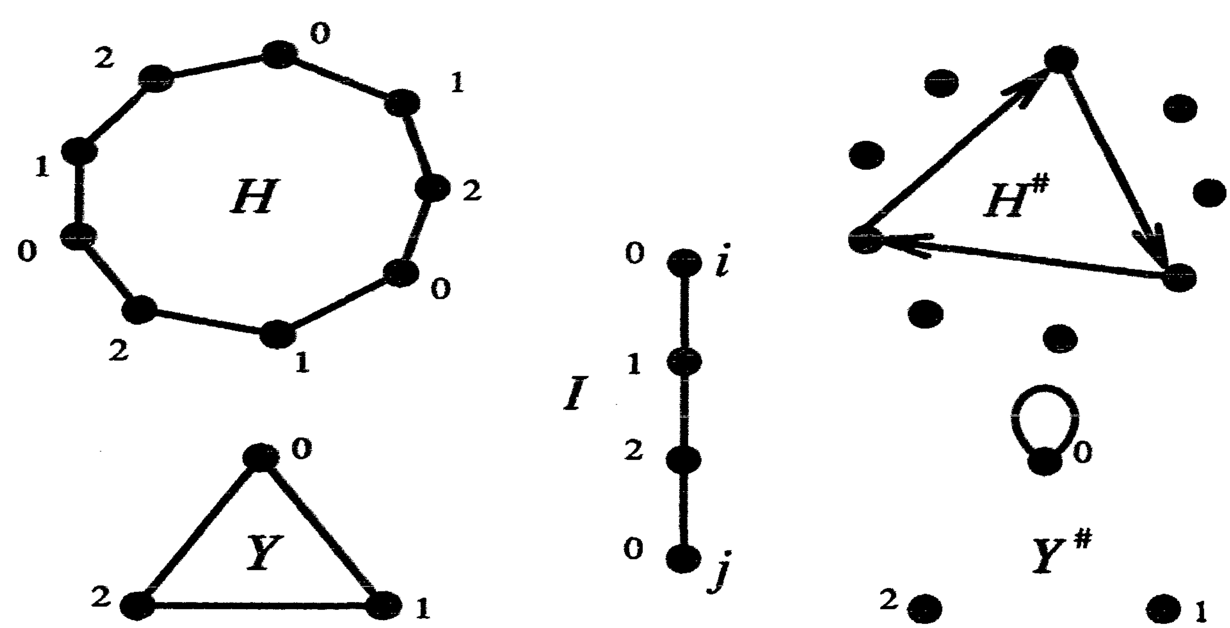


Figure 3.1: An example of HFP indicator construction

Given an arc, $(u, v)$, in $H^{\#}$, there is homomorphism $r : I \to H$ such that $r(i) = u$ and $r(j) = v$ and $h \circ r = t$. Now $t : I \to Y$, $t(i) = (h \circ r)(i) = h(u)$ and $t(j) = (h \circ r)(j) = h(v)$; therefore, $(h(u), h(v))$ is an arc in $Y^{\#}$. That is, $h$ is a homomorphism from $H^{\#}$ to $Y^{\#}$. Hence, $HFP(H^{\#}, h, Y^{\#})$ is a well-defined problem.

**Lemma 3.1.2** *Suppose $H$ and $Y$ are edge-coloured digraphs and $h : H \to Y$ is a homomorphism. Further suppose that $I$ is an edge-coloured digraph with distinguished vertices $i$ and $j$ and that $t : I \to Y$ is a homomorphism. Let $H^{\#}$ and $Y^{\#}$ be the result of the indicator construction with respect to $(I, i, j, t)$. Then $HFP(H^{\#}, h, Y^{\#})$ polynomially transforms to $HFP(H, h, Y)$.*

**Proof.** Let $G, g$ be an instance of $HFP(H^{\#}, h, Y^{\#})$, where $G$ is an edge-coloured, digraph and $g : G \to Y^{\#}$ is a homomorphism. Let ${}^{\#}G$ be the edge-coloured, directed graph obtained by taking a copy of $V(G)$ and for each arc $uv \in E(G)$ putting a copy of $I$ in ${}^{\#}G$ with $i$ identified with $u$ and $j$ identified with $v$. See Figure 3.2 for an

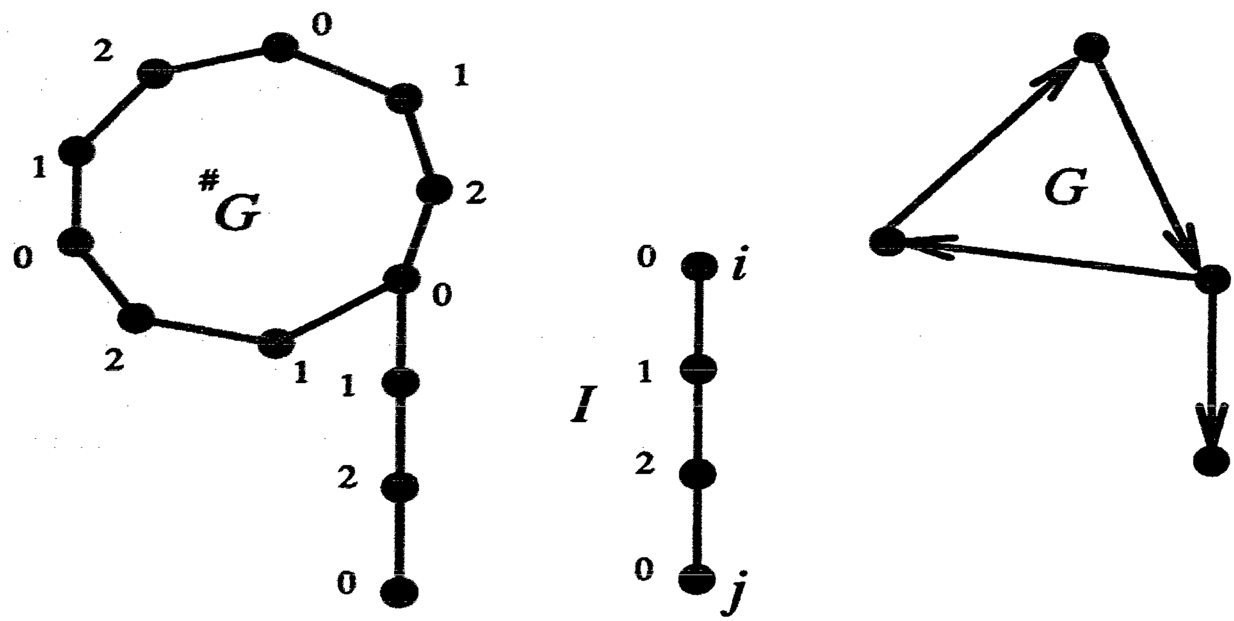


Figure 3.2: The construction of ${}^{\#}G$ from $G$.

example of ${}^{\#}G$. We define a homomorphism ${}^{\#}g : {}^{\#}G \to Y$ as follows:

$$ {}^{\#}g(v) = \begin{cases} g(v) & \text{if } v \in V(G), \\ t(v) & \text{if } v \in V(I)\backslash\{i,j\} \end{cases} $$

It is easy to see that ${}^{\#}g$ is a homomorphism. Finally, we show that $G, g$ is a YES instance of $HFP(H^{\#}, h, Y^{\#})$ if and only if ${}^{\#}G$, ${}^{\#}g$ is a YES instance of $HFP(H, h, Y)$.

Suppose $G, g$ is a YES instance of $HFP(H^{\#}, h, Y^{\#})$, i.e., there exists $f : G \to H^{\#}$ such that $h \circ f = g$. Define a mapping ${}^{\#}f$ from $V({}^{\#}G)$ to $V(H)$ as follows:

- If $u$ is a vertex of $G$, then set ${}^{\#}f(u) = f(u)$.
- If $u$ is not a vertex of $G$, then it must be a vertex in some copy of $I$. This copy of $I$ in ${}^{\#}G$ corresponds to some arc $(x, y)$ in $G$. Since $(f(x), f(y))$ is an arc in $H^{\#}$, there is a homomorphism $r : I \to H$ such that $h \circ r = t$. Set ${}^{\#}f(u) = r(u)$.

Now it is easy to check ${}^{\#}f$ is a homomorphism and $h \circ {}^{\#}f = {}^{\#}g$. The converse is also easy to verify. ∎

We first observe that the HFP problems include $H$-COL.

**Proposition 3.1.3** *Let $H$ be an edge-coloured digraph with multiplicity $k$. Let $Y$ be a digraph which contains a vertex, say $y$, with loops of all colours $1, 2, \dots, k$. Let $h$ be the constant homomorphism $h(v) = y$, for all $v \in V(H)$. Then $HFP(H, h, Y)$ is polynomially equivalent to $H$-COL.*

**Proof.** We begin with a polynomial transformation of $H$-COL to $HFP(H, h, Y)$. Let $G$ be an instance of $H$-COL. We may assume without loss of generality that $G$ is an edge-coloured digraph that uses only edge-colours $1, 2, \dots, k$. Let $g$ be the homomorphism from $G$ to $Y$ defined by $g(v) = y$ for all $v \in V(G)$. It is trivial to check that there is an $f : G \to H$ such that $h \circ f = g$ if and only if $G \to H$. Therefore, $H$-COL $\alpha$ $HFP(H, h, Y)$.

On the other hand, given an instance $G, g$ of $HFP(H, h, Y)$, we consider $G$ as the corresponding instance of $H$-COL. Then $G \to H$ if and only there exists $f : G \to H$ such that $h \circ f = g$. Therefore, $HFP(H, h, Y)$ $\alpha$ $H$-colouring. ∎

We are also able to use the HFP problem to construct polynomial algorithms for certain $H$-colouring problems.

**Theorem 3.1.4** *Let $H$ and $Y$ be edge-coloured digraphs and $h : H \to Y$ a homomorphism such that $HFP(H, h, Y)$ is polynomial. Suppose for any edge-coloured directed graph $G$ the set of homomorphisms from $G$ to $Y$, $\{g : g : G \to Y\}$, can be construct in polynomial time. Then $H$-COL is polynomial.*

**Proof.** We produce a Turing reduction of $H$-COL to $HFP(H,h,Y)$. Let $G$ be an instance of $H$-COL. Construct the set of homomorphisms of $G$ to $Y$, called $Hom(G,Y)$. Let the elements of $Hom(G,Y)$ be $g_1, g_2, \ldots, g_m$. There must be only polynomially many of these homomorphism since the set can be constructed in polynomial time. We claim there exists an $i$ $(1 \leq i \leq m)$ such that $G, g_i$ is a YES instance of $HFP(H,h,Y)$ if and only if $G$ is a YES instance of $H$-COL. On the one hand, the existence of such an $i$ implies there exists $f : G \to H$ such that $h \circ f = g_i$. Trivially, $G$ is a YES instance of $H$-COL. On the other hand, if $G$ is a YES instance of $H$-COL, then there exists $f : G \to H$. The homomorphism $h \circ f : G \to Y$ must be $g_i$ for some $i$. Hence, $G, g_i$ is a YES instance of $HFP(H,h,Y)$. ∎

Given the above proposition and the fact that as yet no complete classification of $H$-colouring for digraphs exists [2], it seems unlikely that we will be able to completely classify the complexity of $HFP(H,h,Y)$ for all $H$, $Y$, and $h : H \to Y$. Therefore, we focus on particular restrictions of $H$, $Y$ and $h$. We begin with a series of results when restrictions on $H$, $h$, and $Y$, give rise to $HFP(H,h,Y)$ problems that are polynomial. The first case we examine is when $h$ is a retraction.

**Lemma 3.1.5** *Let $Y$ be a subgraph of an edge-coloured digraph $H$ and let $h : H \to Y$ be a retraction. Then $HFP(H,h,Y)$ is polynomial.*

**Proof.** Let $G, g$ be an instance of $HFP(H,h,Y)$. The homomorphism $g : G \to Y$ is also a homomorphism of $G$ to $H$ since $Y$ is a subgraph of $H$. Furthermore, $h \circ g = g$ since $h$ is the identity map on $Y$. Therefore, any instance $G, g$ is a YES instance and the problem is trivially solvable in polynomial time. ∎

**Corollary 3.1.6** *Let $Y$ be a subgraph of an edge-coloured digraph $H$ and let $h : H \to Y$ be a retraction of $H$ to $Y$ followed by an automorphism of $Y$. Then $HFP(H,h,Y)$*

*is polynomial.*

**Proof.** Let $h = \sigma \circ h'$ where $h'$ is a retraction of $H$ to $Y$ and $\sigma$ is an automorphism of $Y$. Given an instance $G, g$ of $HFP(H, h, Y)$, it is a YES instance if and only if there exists $f : G \to H$ such that $(\sigma \circ h') \circ f = g$. Since $\sigma$ is an automorphism, it has an inverse, which means the above condition is true if and only if $h' \circ f = \sigma^{-1} \circ g$. Since $\sigma^{-1} \circ g$ is a homomorphism of $G$ to $Y$, this last condition is true if and only if $G, \sigma^{-1} \circ g$ is a YES instance of $HFP(H, h', Y)$.

Hence, $G, g$ is a YES instance of $HFP(H, h, Y)$ if and only if $G, \sigma^{-1} \circ g$ is a YES instance of $HFP(H, h', Y)$. We have $HFP(H, h, Y) \propto HFP(H, h', Y)$. By Lemma 3.1.5, $HFP(H, h, Y)$ is polynomial. ∎

**Corollary 3.1.7** *Let $H$ and $Y$ be edge-coloured digraphs such that $Y$ is the core of $H$. Then for any homomorphism $h : H \to Y$, the problem $HFP(H, h, Y)$ is polynomial.*

**Proof.** Since $Y$ is the core of $H$, any $h : H \to Y$ must be a retraction followed by an automorphism of $Y$. ∎

The following corollary concerns ***homomorphically full*** edge-coloured graphs. These are defined by the property that any homomorphic image is a retract. We discuss homomorphically full graphs in Chapter Eight, see also [7]. Since $HFP(H, h, Y)$ is polynomial when $h$ is a retraction, the following corollary is immediate.

**Corollary 3.1.8** *Let $H$ be a homomorphically full edge-coloured digraph, let $Y$ be any edge-coloured digraph and let $h : H \to Y$ be a homomorphism. Then $HFP(H, h, Y)$ is polynomial.*

We have seen above that placing restrictions on $h$ or on $Y$ can result in a $HFP$ that is polynomial. In general, if no restrictions are placed on $h$ or $Y$ the problem is hard. We examine this in the next section.

## 3.2 Undirected graphs

In this section we restrict our attention to graphs (multiplicity one). We may assume for all $HFP(H, h, Y)$ problems, that the graph $Y$ is connected. We prove the following main result.

**Theorem 3.2.1** *For each connected graph $Y$, $Y \notin \{P_0, P_1, P_2, P_3\}$, there exists a graph $H$ and a homomorphism $h : H \to Y$ such that $HFP(H, h, Y)$ is NP-complete. For each graph $Y \in \{P_0, P_1, P_2, P_3\}$ and for all graphs $H$ and all homomorphisms $h : H \to Y$, the problem $HFP(H, h, Y)$ is polynomial.*

**Proof.** First suppose $Y$ is a graph and is not one of $\{P_0, P_1, P_2, P_3\}$. Depending on $Y$, we will choose a graph $H$ and a homomorphism $h : H \to Y$ so that $HFP(H, h, Y)$ is NP-complete.

**Case 1: Suppose $Y$ contains a cycle.** Let $C = c_0c_1 \ldots c_{n-1}$ be a cycle in $Y$. Let $H$ be the graph consisting of two cycles, of length $3n$ and $4n$ respectively, joined at a single vertex. Label the vertices in these two cycles with $v_0v_1v_2 \ldots v_{3n-1}$ and $u_0u_1u_2 \ldots u_{4n-1}$, where $v_0 = u_0$. Let $h : H \to Y$ be the homomorphism defined by:

$$\left.\begin{array}{r} h(v_i) \\ h(u_i) \end{array}\right\} = c_j \text{ where } i \equiv j \bmod n.$$

Let $I$ be a path of length $n$ on vertices $p_0p_1 \dots p_n$ with $i = p_0$ and $j = p_n$. Define $t : I \to Y$ as follows:

$$\begin{aligned} t(p_i) &= c_i \text{ for } 0 \le i < n, \\ t(p_n) &= c_0. \end{aligned}$$

The result of the indicator construction with respect to $(I, i, j, t)$ is $H^{\#}$, $Y^{\#}$. The digraph $H^{\#}$ consists of a directed three-cycle and a directed four-cycle joined at the vertex $v_0$ (plus isolated vertices). In particular, the vertices of the three-cycle are $v_0, v_n, v_{2n}$ and the vertices of the four-cycle are $v_0 = u_0, u_n, u_{2n}, u_{3n}$. It is important to note that the choice of the homomorphism $h$, here and below, assures that $H^{\#}$ contains no other edges and in particular no loops. The digraph $Y^{\#}$ has a single arc, namely a loop on vertex $c_0$. Also the map $h(v) = c_0$ for all $v \in V(H^{\#})$ is a homomorphism of $H^{\#}$ to $Y^{\#}$. By Proposition 3.1.3, we have $HFP(H^{\#}, h, Y^{\#})$ is polynomially equivalent to $H^{\#}$-COL. In [2], it is shown that $H$-COL is NP-complete when $H$ consists of two directed cycles joined at a single vertex (assuming $H$ does not retract to a single cycle). Hence by [2], $H^{\#} - COL$ is NP-complete and therefore $HFP(H, h, Y)$ is NP-complete.

**Case 2: Suppose that $Y$ contains a vertex of degree at least three.** Let $y$ be a vertex in $Y$ with neighbours $u, v, w$. Let $I$ be the path $p_0p_1p_2 \dots p_6$ with $i = p_0$ and $j = p_6$. Let $t : I \to Y$ be defined by:

$$t(p_0) = t(p_2) = t(p_4) = t(p_6) = y$$

$$\begin{aligned} t(p_1) &= u \\ t(p_3) &= v \\ t(p_5) &= w \end{aligned}$$

We now proceed as above. Let $H$ be the graph consisting of two cycles of lengths 18 and 24 joined at a single vertex. Let the first cycle have vertex-set $\{c_0, c_1, \ldots, c_{17}\}$ and the second have vertex-set $\{d_0, d_1, \ldots, d_{23}\}$, where $d_0 = c_0$. Define $h : H \to Y$ by:

$$h(c_i) = h(d_i) = \begin{cases} y & i \equiv 0, 2, 4 \pmod 6, \\ u & i \equiv 1 \pmod 6, \\ v & i \equiv 3 \pmod 6, \\ w & i \equiv 5 \pmod 6. \end{cases}$$

The result of the indicator construction with respect to $(I, i, j, t)$ is $H^{\#}$ and $Y^{\#}$. The digraph $H^{\#}$ consists of a directed three-cycle and a directed four-cycle joined at a single vertex. The graph $Y^{\#}$ has a loop on vertex $y$. The homomorphism $h : H^{\#} \to Y^{\#}$ defined by $h(z) = y$ for all $z \in V(H^{\#})$ satisfies the condition of Proposition 3.1.3. Therefore, $HFP(H^{\#}, h, Y^{\#})$ is equivalent to $H^{\#}$-colouring. Again, this is NP-complete. Hence, $HFP(H, h, Y)$ is NP-complete.

**Case 3: Suppose $Y$ is a path of length at least four.** Label the first five vertices of $Y$ with $0, 1, 2, 3, 4$. Let $I$ be a path on 13 vertices, $p_0, p_1, \ldots, p_{12}$ with $i = p_0$ and $j = p_{12}$. Define a homomorphism $t : I \to Y$ so that the vertices of $I$ have consecutively the images 0, 1, 2, 1, 2, 3, 4, 3, 2, 3, 2, 1, 0. As above, $H$ is a graph consisting of two cycles joined at a vertex. One cycle is constructed by taking three copies of $I$ and identifying $j$ of the first copy with $i$ of the second, $j$ of the second with $i$ of the third, and $j$ of the third with $i$ of the first. The second cycle consists of four copies of $I$ and is constructed similarly. Join the two cycles by identifying the vertex $i$ in the first copies of $I$ in each of the cycles. The homomorphism $h$ is correspondingly constructed from the homomorphisms $t$ of the individual copies of $I$. The result of the indicator construction with respect to $(I, i, j, t)$ is $H^{\#}$, $Y^{\#}$, where $H^{\#}$ and $Y^{\#}$

are as above. By Proposition 3.1.3, $HFP(H, h, Y)$ is NP-complete.

If a connected $Y$ does not contain a cycle or a vertex of degree at least three, it must be a path. If $Y$ is not a path of length at least four, it must be one of $\{P_0, P_1, P_2, P_3\}$. Therefore, the three cases above prove the first part of the theorem. The second half of the theorem is proved below.

Hence, suppose that $Y \in \{P_0, P_1, P_2, P_3\}$. Let $H$ be a graph. Recall that we can always assume $H$ is connected. If $Y = P_0$ and $h : H \to Y$ is a homomorphism, then $h$ is a retraction and by Lemma 3.1.5 $HFP(H, h, Y)$ is polynomial. If $Y = P_1$ and $h : H \to Y$ is a homomorphism, then we can conclude $h$ is a retraction followed by an automorphism (and hence $HFP(H, h, Y)$ is polynomial) unless $h$ is not onto. If this is the case, then $h$ must map $H$ onto $P_0$ and $HFP(H, h, P_1)$ is equivalent to $HFP(H, h, P_0)$ and hence also polynomial. If $Y = P_2$ and $h$ is a homomorphism from $H$ to $Y$, then again $HFP(H, h, Y)$ is polynomial if $h$ is onto, because $h$ is a retraction followed by an automorphism; otherwise, $h$ is not onto and $HFP(H, h, Y)$ is equivalent to $HFP(H, h, P_0)$ or $HFP(H, h, P_1)$ and hence also polynomial.

Finally suppose $Y$ is $P_3$ and $h$ is a homomorphism $h : H \to Y$. Let the vertex-set of $Y$ be $\{0, 1, 2, 3\}$. If $h$ is not onto, then $HFP(H, h, Y)$ is equivalent to one of the three polynomial problems above. Hence, assume $h$ is onto. Let $P = (v = p_0)p_1 \ldots (p_n = u)$ be a shortest path in $H$ from a vertex $v$, such that $h(v) = 0$, to a vertex $u$, such that $h(u) = 3$. Since $P$ is a shortest path, no interior vertex of $P$ is mapped to 0 or to 3. The vertices in $P$ have the consecutive images under $h$: $0, 1, 2, 1, 2, 1, 2, \ldots, 1, 2, 3$. It is easy to check that there is a retraction $r$ from $H$ to $P$ such that $h \circ r = h$. Given an instance $G, g$ of $HFP(H, h, Y)$ we can construct a shortest path, say $Q$, between

$u$ and $v$ where the $u$ and $v$ are taken over all pairs $u$ and $v$ such that $g(u) = 0$ and $g(v) = 3$. This problem is the Shortest Pairs Problem and is polynomial [12]. Again it is easy to see there is a retraction $t$ from $G$ to $Q$ such that $g \circ t = g$. Finally, $G, g$ is a YES instance of $HFP(H, h, Y)$ if and only if there exists $f : Q \rightarrow P$ such that $h \circ f = g$. This is true if and only if the length of $Q$ is greater than or equal to the length of $P$. ∎

Theorem 3.2.1 deals with graphs. However, the algorithm stated in the final case works for edge-coloured digraphs. This gives an immediate corollary.

**Corollary 3.2.2** *Let $Y$ be an edge-coloured directed path of length 0, 1, 2, or 3. Then for all edge-coloured digraphs $H$ and all homomorphisms $h : H \rightarrow Y$, the problem $HFP(H, h, Y)$ is polynomial.*

Using the above corollary and Theorem 3.1.4 we have a new class of $H$-colouring problems that are all polynomial.

**Corollary 3.2.3** *Let $H$ be an edge-coloured digraph such that $H$ admits a homomorphism to an edge-coloured directed path of length at most three. Then $H$-COL is polynomial.*

**Proof.** Let $Y$ be an edge-coloured directed path of length $k$, where $k \leq 3$, and let $h : H \rightarrow Y$ be a homomorphism. By the above corollary, $HFP(H, h, Y)$ is polynomial. It is easy to check that given any edge-coloured digraph $G$, there exist at most $k$ homomorphisms from $G$ to $Y$. Since $Y$ is a fixed path, $k$ is a constant and therefore the number of homomorphisms of $G$ to $Y$ is bounded by the constant polynomial $k$. Using Theorem 3.1.4, $H$-COL is polynomial. ∎

## 3.3 HFP as Edge-Coloured H-colouring

In this section we again consider the HFP problem for graphs, although there is a natural generalization to edge-coloured graphs. Consider $HFP(H, h, Y)$ and an instance $G, g$ of it. We now present a construction that transforms the graphs $G$, $H$ and the homomorphisms $g$, $h$ into edge-coloured digraphs $G_c$, $H_c$, such that $G, g$ is a YES instance of $HFP(H, h, Y)$ if and only if $G_c$ admits a homomorphism to $H_c$.

Suppose $Y$ and $H$ are graphs and $h : H \to Y$ is a homomorphism. Let $V(Y) = \{y_0, y_1, \ldots, y_k\}$ and let $C$ be the set of all unordered pairs of elements of $V(Y)$; the set $C$ is the set of edge-colours of our new digraph. We construct the edge-coloured digraph $H_c$ as follows:

- $V(H_c) = V(H)$;
- for each edge $uv$ of $H$ where $h(u) = y_i$ and $h(v) = y_j$ and $i < j$, the arc $uv$ is an edge of colour $\{y_i, y_j\}$ in $H_c$.

Let $G$, $H$, and $Y$ be graphs and let $h : H \to Y$ and $g : G \to Y$ be homomorphisms. It is easy to check that $G_c$ admits a homomorphism to $H_c$ if and only there exists a homomorphism $f : G \to H$ such that $h \circ f = g$. That is, if and only if $G, g$ is a YES instance of $HFP(H, h, Y)$. This gives the following proposition.

**Proposition 3.3.1** *Let $H$, $Y$, $h$, and $H_c$ be defined as above. Then $HFP(H, h, Y)$ $\alpha$ $H_c$-COL.*

Two immediate corollaries to the proposition are given below.

**Corollary 3.3.2** *Let $H$ be a path, $Y$ be a graph and $h : H \to Y$ a homomorphism. Then $HFP(H, h, Y)$ is polynomial.*

**Proof** The edge-coloured digraph $H_c$ an oriented path. Testing for the existence of a homomorphism to an oriented path is polynomial, [14], i.e. $H_c$-COL is polynomial. ∎

**Corollary 3.3.3** *Let $H$ and $Y$ be graphs and $h : H \to Y$ a homomorphism. Further suppose for all $u \in V(H)$, $h$ is one-to-one on the neighbourhood of $u$. Then $HFP(H, h, Y)$ is polynomial.*

**Proof** Since $h$ is one-to-one on the neighbourhood of each vertex of $H$, $H_c$ has at most one arc of each colour incident with any given vertex. Therefore, by the Forcing Lemma (Lemma 2.2.3), $H_c$-COL is polynomial and hence $HFP(H, h, Y)$ is polynomial. ∎

## 3.4 The Two Homomorphism Problem

We now examine a problem similar to HFP. Again we restrict our attention to multiplicity one. In the spirit of the HFP we consider the problem when $Y$ is fixed and $H = G$ is part of the instance. Formally,

Let $Y$ be a fixed graph:

**Two Homomorphism Problem** $THP(Y)$.

INSTANCE: A graph $H$ and a two homomorphisms $h_1 : H \to Y$ and $h_2 : H \to Y$.

QUESTION: Does there exist a homomorphism $f : H \to H$ such that $h_1 \circ f = h_2$?

In the proof of Theorem 3.2.1 the result of the indicator construction, $H^{\#}$, is always a directed three-cycle joined at a vertex to a directed four-cycle. In $H$, these cycles correspond to three copies of the indicator $I$ and four copies of the indicator $I$. If instead the graph $H$ consisted of four copies of $I$ and six copies of $I$, the result of the indicator construction $H^{\#}$ would be a four-cycle joined at a vertex to a six-cycle. Therefore, the "4-cycle-6-cycle" version of the proof gives the following corollary.

**Corollary 3.4.1** *Let $Y$ is be a connected graph and suppose that $Y \notin \{P_0, P_1, P_2, P_3\}$. Then there exists a bipartite graph $H$ and a homomorphism $h : H \to Y$ such that $HFP(H, h, Y)$ is NP-complete.*

We now state our main result for the Two Homomorphism Problem.

**Theorem 3.4.2** *Let $Y$ a connected graph and $Y \notin \{P_0, P_1, P_2, P_3\}$. Then $THP(Y)$ is NP-complete.*

**Proof.** By Corollary 3.4.1, there exists a bipartite graph $H$ and a homomorphism $h : H \to Y$ such that $HFP(H, h, Y)$ is NP-complete. We find a polynomial transformation of $HFP(H, h, Y)$ to $THP(Y)$. Let $G, g$ be an instance of $HFP(H, h, Y)$.

The graph $G$ must be bipartite in order to admit a homomorphism to $H$. Therefore, if $G$ is not bipartite, then $G, g$ is a NO instance of $HFP(H, h, Y)$. Thus, assume $G$ is bipartite. Also, to avoid the trivial case, assume $G$ contains at least one edge.

Let $y_0y_1$ be an edge of $Y$. We begin by examining two special cases. First, if $g(G)$ is $y_0y_1$, then $G, g$ is a YES instance of $HFP(H, h, Y)$ if and only if $H$ contains an edge $uv$ such that $h(u) = y_0$ and $h(v) = y_1$ Second, if $h(H)$ is $y_0y_1$, then $G, g$ is a YES instance of $HFP(H, h, Y)$ if and only if $g(G)$ is the edge $y_0y_1$. Therefore, assume neither $g(G)$ nor $h(H)$ is $y_0y_1$.

Let $H'$ be the union of $G$ and $H$. Let $f_1$ be the homomorphism that maps $H$ to $y_0y_1$ and is equal to $g$ on $G$. Similarly, let $f_2$ be the homomorphism that maps $G$ to $y_0y_1$ and is equal to $h$ on $H$. The instance $H', f_1, f_2$ is a YES instance of $THP(Y)$ if and only if $G, g$ is a YES instance of $HFP(H, h, Y)$.

Suppose $G, g$ is a YES instance of $HFP(H, h, Y)$. This implies there is $f : G \to H$ such that $h \circ f = g$. Let $t$ be the homomorphism $H'$ to $H'$ defined by:

$$t(u) = \begin{cases} f(u) & \text{if } u \in V(G) \\ f_1(u) & \text{if } u \in V(H) \end{cases}$$

It is easy to check that $f_2 \circ t = f_1$.

On the other hand, suppose $H', f_1, f_2$ is a YES instance of $THP(Y)$. Let $f : H' \to H'$ be a homomorphism such that $f_2 \circ f = f_1$. Consider $f$ restricted to $G$. This is a homomorphism from $G$ to $H'$. Since we can assume $G$ is connected and $H'$ consists of the connected components $G$ and $H$, either $f(G)$ is a subgraph of $G$ or $f(G)$ is a subgraph of $H$. Since $f_1(G)$ is not $y_0y_1$ and $f_2(G)$ is $y_0y_1$, it must be the case that $f(G)$ is contained in $H$. By restricting $f$ to $G$, $f_1$ to $G$ and $f_2$ to $H$, we have $h \circ f = g$. Therefore, $G, g$ is a YES instance of $HFP(H, h, Y)$. ■

## 3.5 The Bipartite Decomposition Lemma

The next technique allows us to take a given $H$-colouring problem and split it into to two smaller problems $H_1$-COL and $H_2$-COL. The complexity of these problems determines the complexity of $H$-COL. Suppose $H$ is an edge-coloured graph which is a core and blue is an edge-colour of $H$. Further suppose that the blue spanning subgraph is a complete bipartite graph with bipartition $(A, B)$. Moreover, suppose that for all $u \in A$ and $v \in B$, $uv$ is not an edge of any colour except blue. Let $H_1$ be the induced subgraph of $H$ with vertex-set $A$ and $H_2$ be the induced subgraph of $H$ with vertex-set $B$. We have the following lemma.

**Lemma 3.5.1** *Let $H$, $H_1$, and $H_2$ be as above. If $H_1$-COL or $H_2$-COL is NP-complete, then $H$-COL is NP-complete. If both $H_1$-COL and $H_2$-COL are polynomial, then $H$-COL is polynomial.*

**Proof.** We prove the former statement first. Suppose $H_1$-COL is NP-complete. Let $h_1$ be a vertex in $H_1$. Let $J$ be a blue path of length two with $j$ at one end and $k_1$ at the other. The result of the subindicator construction on $H$ with respect to $J, j, k_1$ is $H_1$. By assumption this problem is NP-complete and hence $H$-COL is NP-complete. A similar argument works when $H_2$-COL is NP-complete.

Now suppose both $H_1$-COL and $H_2$-COL are polynomial. Let $C$ be the congruence with two classes $S_1 = V(H_1) = A$ and $S_2 = V(H_2) = B$. Let $H'$ be the quotient of this congruence and let $h$ be the homomorphism from $H$ to $H'$ induced by the congruence. It is easy to check that $HFP(H, h, H')$ is polynomial. Furthermore, $H'$ has the property that each vertex is incident with one blue edge and loops of several colours, but is not incident with a blue loop. Therefore, $H'$ satisfies the hypothesis of Lemma 2.2.3. Hence, by Theorem 3.1.4, $H$-COL is polynomial. ∎

In the first part of the proof, we did not use the fact that only blue edges pass from $H_1$ to $H_2$. That is, the sub-indicator construction works whenever the blue edges induce a complete bipartite graph regardless of the other edges between $H_1$ and $H_2$. Hence, we have the following corollary.

**Corollary 3.5.2** *Let $H$ be an edge-coloured graph such that the blue spanning subgraph is a complete bipartite graph with bipartition $(A, B)$. Let $H_1$ (resp. $H_2$) be the subgraph induced by $A$ (resp. $B$). If $H_1$-COL or $H_2$-COL is NP-complete, then $H$-COL is NP-complete.*

# Chapter 4

# Bipartite Two-Edge-Coloured Graphs

## 4.1 Equivalence to Directed Graphs

The $H$-colouring problems for two-edge-coloured bipartite graphs and for bipartite digraphs turn out to be closely related. In this section we show how to construct a bipartite digraph from a given bipartite two-edge-coloured graph and vice versa so that the corresponding $H$-colouring problems are polynomially equivalent. In this chapter we make the assumption that all edge-coloured graphs and digraphs are connected.

Let $H$ be a two-edge-coloured graph. Define the *converse of* $H$, written $H^R$, to be the edge-coloured graph on vertex $V(H)$, where $E_1(H) = E_2(H^R)$ and $E_2(H) = E_1(H^R)$. That is, $H^R$ is obtained from $H$ by interchanging red and blue edges. Let $D$ be a directed graph. Define the *converse of* $D$, written $D^R$, to be the directed graph on vertex-set $V(D)$ where $uv \in E(D)$ if and only if $vu \in E(D^R)$. The following proposition is straightforward and is presented without proof.

**Proposition 4.1.1** *Suppose $G$ and $H$ are two-edge-coloured graphs. Then $G \to H$ if and only if $G^R \to H^R$.*

Similarly we can show for the following for digraphs.

**Proposition 4.1.2** *Suppose $C$ and $D$ are digraphs. Then $C \to D$ if and only if $C^R \to D^R$.*

We now explain how to construct a bipartite digraph from a bipartite two-edge-coloured graph. Let $H$ be a bipartite edge-coloured graph and $(H_0, H_1)$ a bipartition of $H$. Define $Dir(H, H_0, H_1)$ to be the directed graph $D$ as follows:

- Let $V(D) = V(H)$,
- let $uv \in E(D)$ for all $u \in H_0$, $v \in H_1$, and $uv \in E_1(H)$,
- let $vu \in E(D)$ for all $u \in H_0$, $v \in H_1$, and $uv \in E_2(H)$.

Briefly, $D$ is the digraph obtained by replacing each blue edge in $H$ from $H_0$ to $H_1$ by a forward arc and each red edge in $H$ from $H_0$ to $H_1$ by a backward arc. Note that if $(H_0, H_1)$ is a bipartition of $H$, then $(H_1, H_0)$ is also a bipartition of $H$. Moreover, if $(H_0, H_1)$ is a bipartition of $H$, then it is also a bipartition of $Dir(H, H_0, H_1)$. The following proposition is straightforward.

**Proposition 4.1.3** *Let $H$ be a bipartite two-edge-coloured graph . Suppose that $(H_0, H_1)$ is a bipartition of $H$ and $D = Dir(H, H_0, H_1)$. Then $D^R = Dir(H, H_1, H_0)$.*

We also have a construction to construct a bipartite edge-coloured graph from a bipartite digraph. Let $D$ be a bipartite digraph $D$ and $(D_0, D_1)$ a bipartition of $D$. Define $ECG(D, D_0, D_1)$ to be the edge-coloured graph $H$ as follows:

- Let $V(H) = V(D)$,
- let $uv \in E_1(H)$ for all $u \in D_0$, $v \in D_1$, and $uv \in E(D)$,
- let $uv \in E_2(H)$ for all $u \in D_0$, $v \in D_1$, and $vu \in E(D)$.

Briefly, $H$ is the edge-coloured graph obtained by replacing each arc from $D_0$ to $D_1$ by a blue edge and each arc from $D_1$ to $D_0$ by a red edge. We now present the analogous result to Proposition 4.1.3.

**Proposition 4.1.4** *Suppose $D$ is a digraph. Suppose that $(D_0, D_1)$ is a bipartition of $D$ and $H = ECG(D, D_0, D_1)$. Then $H^R = ECG(D, D_1, D_0)$.*

The above constructions preserve edge structure in some sense and hence preserve homomorphisms. This is described in the following theorem.

**Theorem 4.1.5** *Let $G$ and $H$ be bipartite two-edge-coloured graphs with bipartitions $(G_0, G_1)$ and $(H_0, H_1)$ respectively. Let $C = Dir(G, G_0, G_1)$ and $D = Dir(D, H_0, H_1)$. Then $G \to H$ or $G^R \to H$ if and only if $C \to D$ or $C^R \to D$.*

**Proof** Suppose $G \to H$ or $G^R \to H$. We show $C \to D$ or $C^R \to D$. Assume there is a homomorphism $f : G \to H$. The case $G^R \to H$ is similar. Either $f(G_0) \subseteq H_0$ and $f(G_1) \subseteq H_1$ or $f(G_0) \subseteq H_1$ and $f(G_1) \subseteq H_0$.

First, suppose the former case holds. To see that $f : C \to D$ is a homomorphism, let $uv \in E(C)$. If $u \in C_0$ and $v \in C_1$, then $uv \in E_1(G)$. Hence $f(u)f(v) \in E_1(H)$. This implies $f(u)f(v) \in E(D)$, since $f(u) \in H_0$ and $f(v) \in H_1$. On the other hand, if $u \in C_1$ and $v \in C_0$, then $uv \in E_2(G)$ which implies $f(u)f(v) \in E_2(H)$ and $f(u)f(v) \in E(D)$. In both cases $f(u)f(v) \in E(D)$.

Second, suppose the latter case holds. Then $f : C \to D^R$ is a homomorphism. Let $uv \in E(C)$. If $u \in C_0$ and $v \in C_1$, then $uv \in E_1(G)$. Hence $f(u)f(v) \in E_1(H)$. We are now assuming $f(u) \in H_1$ and $f(v) \in H_0$. Hence, $f(v)f(u) \in E(D)$ and $f(u)f(v) \in E(D^R)$. On the other hand, if $u \in C_1$ and $v \in C_0$, then $uv \in E_2(G)$ which implies $f(u)f(v) \in E_2(H)$. Therefore, $f(v)f(u) \in E(D)$ and $f(u)f(v) \in E(D^R)$. In both cases $f(u)f(v) \in E(D^R)$.

Thus we can conclude $C \to D$ or $C \to D^R$. The proof that $C \to D$ or $C \to D^R$ implies $G \to H$ or $G \to H^R$ is similar. ■

The above theorem suggests that for $H$ a bipartite two-edge-coloured graph and $D$ a bipartite digraph, $H$-COL and $D$-COL are polynomially equivalent. This is in fact proved below. Since the edge-coloured graph $H = ECG(Dir(H, H_0, H_1), H_0, H_1)$ and the digraph $D = Dir(ECG(D, D_0, D_1), D_0, D_1)$, the following theorem can be stated in two versions. Namely, we can consider $D$ to be constructed from $H$ or $H$ to be constructed from $D$. By the previous observation they are equivalent.

**Theorem 4.1.6** *Suppose $H$ is a bipartite two-edge-coloured graph. Suppose that $(H_0, H_1)$ is a bipartition of $H$ and $D = Dir(H, H_0, H_1)$. Then $H$-COL $\alpha_T$ $D$-COL and $D$-COL $\alpha_T$ $H$-COL.*

**Proof** Let $G$ be an instance of $H$-COL. Choose a specific vertex $v$ of $G$. Let $h_1, h_2, \ldots, h_k$ be the vertices of $H$. Define $G_i$ as the edge-coloured graph obtained from the disjoint union of $G$ and $H$, when vertex $v \in V(G)$ and vertex $h_i \in V(H)$ are identified.

**Claim 4.1.6.1** *The edge-coloured graph $G$ admits a homomorphism to $H$ if and only if for some $i$ the edge-coloured graph $G_i$ admits a homomorphism to $H$.*

Suppose $g : G_i \to H$ is a homomorphism. Since $G$ is a subgraph of $G_i$, the existence of $g$ implies the existence of $f : G \to H$. On the other hand, suppose $f : G \to H$ is a homomorphism and $f(v) = h_i$. Define $g$ as:

$$g(u) = \begin{cases} f(u) & \text{if } u \in V(G) \\ u & \text{if } u \in V(H) \end{cases}$$

The verification that $g$ is a homomorphism from $G_i$ to $H$ is immediate. The claim follows.

A second observation about the sequence $G_i$ is given in the following claim.

**Claim 4.1.6.2** *Suppose $G_i \to H^R$. Then $G_i \to H$.*

Suppose $f : G_i \to H^R$ is a homomorphism. Since $H$ is a subgraph of $G_i$, the existence of $f$ implies $H \to H^R$. By Proposition 4.1.1 it must be the case that there exists a homomorphism $h : H^R \to (H^R)^R$. However, $(H^R)^R$ is simply $H$. Therefore $h \circ f$ is a homomorphism from $G_i$ to $H$. This proves the claim.

An immediate consequence of this claim is that:

$$G_i \to H \text{ if and only if } G_i \to H \text{ or } G_i \to H^R$$

Combining this observation with the first claim we see that:

$$G \to H \text{ if and only if there exists } i \text{ such that } G_i \to H \text{ or } G_i \to H^R$$

For each $i$ define $C_i = Dir(G_i, A, B)$ where $(A, B)$ is a bipartition of $G_i$. Recall that $H$ is bipartite and any edge-coloured graph that maps to $H$ must also be bipartite; hence, $G_i$ is bipartite. Since $C_i \to D^R$ if and only if $C_i^R \to D$, the following is true:

$$C_i \to D \text{ or } C_i^R \to D \text{ if and only if } C_i \to D \text{ or } C_i \to D^R$$

Combining these observations with Theorem 4.1.5 we have:

$$\begin{aligned} G \to H &\Leftrightarrow \text{there exits } i \text{ such that } G_i \to H \\ &\Leftrightarrow \text{there exits } i \text{ such that } G_i \to H \text{ or } G_i \to H^R \\ &\Leftrightarrow \text{there exits } i \text{ such that } C_i \to D \text{ or } C_i \to D^R \\ &\Leftrightarrow \text{there exits } i \text{ such that } C_i \to D \text{ or } C_i^R \to D \end{aligned}$$

This final expression can be evaluated as it is $2k$ instances of $D$-COL. Hence $H$-COL $\alpha_T$ $D$-COL. The converse is proved is a similar way. ■

## 4.2 Consequences

The following list of propositions follows from Theorem 4.1.6 and the literature on $H$-COL for digraphs.

In [14] the authors show that $H$-COL is polynomial for oriented paths; however, there exists an oriented tree on 288 vertices such that $H$-COL is NP-complete. This implies that $H$-COL is polynomial for two-edge-coloured paths. It also implies the existence of a two-edge-coloured tree for which $H$-COL is NP-hard. In this thesis we present a two-edge-coloured tree on 98 vertices for which $H$-COL is NP-complete; moreover, the tree is structurally simple in that it contains a unique vertex of degree greater than two. This provides an oriented tree, smaller and simpler, than the tree in [14].

In [2] the authors show that for complete bipartite digraphs, the $H$-COL problem is NP-hard if $H$ contains two directed cycles and is polynomial otherwise. This implies the following proposition.

**Proposition 4.2.1** *Let $H$ be a two-edge-coloured complete bipartite graph. If $H$ contains two cycles whose edges alternate red and blue, then $H$-COL is NP-hard. Otherwise, $H$-COL is polynomial.*

The final observations involve cycles. An even length oriented cycle can be transformed into an even length two-edge-coloured cycle and vice versa. In [30] and [15] the authors have independently shown for any oriented cycle, $C$, containing more forward arcs than backward arcs, $C$-COL is polynomial. They also show there exists oriented cycles for which $C$-COL is NP-complete; these cycles must have the same number of forward arcs and backward arcs and therefore have even length. The implications for edge-coloured graphs are the existence of polynomial algorithms for some two-edge-coloured cycles and the existence of NP-hard two-edge-coloured bipartite cycles. This is further discussed in Chapter Six.

# Chapter 5

# Path and Tree Colourings

## 5.1 Path Colourings

In this section we study the complexity of $H$-COL when $H$ is a fixed, edge-coloured path. We present a polynomial algorithm to solve this problem. Recall that $G \to H$ if and only if $G \to G \times H$ (Lemma 1.1.2). We shall give an algorithm to decide whether or not $G \to G \times H$; the idea of our algorithm is similar to the algorithm for the uncoloured case, see [14]. In this chapter we restrict our attention to loop-free edge-coloured graphs; however, we put no restriction on the number of edge-colours.

For this entire section assume all edge-coloured graphs $G$ and $H$ are bipartite since we ultimately wish to solve $H$-COL where $H$ is a path and $G$ is an instance of the problem. Note that if $H$ is a path and $G$ is not bipartite, we can answer NO to $H$-COL, since any preimage of a path must be bipartite. We begin with a series of lemmas specific to bipartite edge-coloured graphs.

**Lemma 5.1.1** *Let $G$ and $H$ be bipartite edge-coloured graphs and $(g_1, h_1)(g_2, h_2)$ an edge of $G \times H$. Then $(g_2, h_1)(g_1, h_2)$ is also an edge of $G \times H$ and these two edges lie in different components of $G \times H$.*

**Proof.** Let $(G_1, G_2)$ (resp. $(H_1, H_2)$) be a partition of the vertices of $G$ (resp. $H$) into two independent sets.

Let $(g_1, h_1)(g_2, h_2) \in E_i(G \times H)$ for some $i$. Observe that $g_1g_2 \in E_i(G)$ and $h_1h_2 \in E_i(H)$; hence, by the definition of $G \times H$, $(g_2, h_1)(g_1, h_2)$ is also an edge of $G \times H$. Furthermore, $g_1$ and $g_2$ must be in different parts of the partition of $G$ and similarly $h_1$ and $h_2$ must be different parts of the partition of $H$. All edges in $G \times H$ either have one end in $G_1 \times H_1$ and the other end in $G_2 \times H_2$ or have one end in $G_1 \times H_2$ and the other end in $G_2 \times H_1$. Therefore, $(g_1, h_1)(g_2, h_2)$ and $(g_1, h_2)(g_2, h_1)$ are in different components of $G \times H$. ∎

**Corollary 5.1.2** *If $G \times H$ is not a single vertex, then it has at least two components.*

Note it is possible to have more than two components in $G \times H$ even when $G$ and $H$ are connected edge-coloured graphs as shown in Figure 5.1. This differs from (classical) graphs. (See Proposition 1 [28].)

The algorithm in [14] solves the $H$-COL problem in polynomial time when $H$ is an oriented (uncoloured) path. We present a similar algorithm for the case when $H$ is an edge-coloured path. However, the algorithm in [14] requires that the target graph $H$ have the so-called $\underline{X}$ property. The edge-coloured graphs we study do not have the $\underline{X}$ property. Instead we use our lemmas to show that "crossing" edges in $G \times H$ are in different components. That is, each component of $G \times H$ has the $\underline{X}$ property. Our

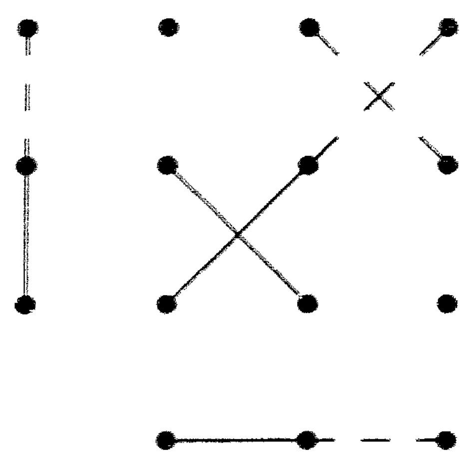

Figure 5.1: A product of two connected, edge-coloured graphs with five components.

result can also be derived using the notion of $C_k$-extension of the $\underline{X}$ property found in [14].

We now assume $H$ is an edge-coloured path. Let $n_g = |V(G)|$ and $n_h = |V(H)|$. For the following we will assume the vertices of $H$ have been labeled with $\{0, 1, 2, \ldots, n_h - 1\}$ so that $i(i+1) \in E(H)$ for $0 \leq i \leq n_h - 2$. For convenience we will arbitrarily label $V(G)$ with $\{0, 1, 2, \ldots, n_g - 1\}$.

**Lemma 5.1.3** *Let $(g_1, h_1)(g_2, h_2)$ and $(g_1, h_3)(g_2, h_4)$ be two distinct edges in the same component of $G \times H$, then ($h_1 \leq h_3$ and $h_2 \leq h_4$) or ($h_1 \geq h_3$ and $h_2 \geq h_4$).*

**Proof.** We can assume without loss of generality that $h_1 \leq h_2$. Since $H$ is a path and $h_1h_2$ is an edge of H, $h_2 = h_1 + 1$. Suppose $h_3 \leq h_4$. Again, since $H$ is a path, $h_4 = h_3 + 1$. If $h_1 \leq h_3$, then $h_2 \leq h_4$. If $h_3 \leq h_1$, then $h_4 \leq h_2$. In either case the result holds. Now suppose $h_3 \geq h_4$, i.e. $h_3 = h_4 + 1$. If $h_3 \leq h_1$ or $h_4 \geq h_2$, then the result is true. Hence, the only way for the lemma to fail is if $h_3 > h_1$ and $h_4 < h_2$. This implies $h_4 = h_1$ and $h_3 = h_2$. By Lemma 5.1.1 these edges are in

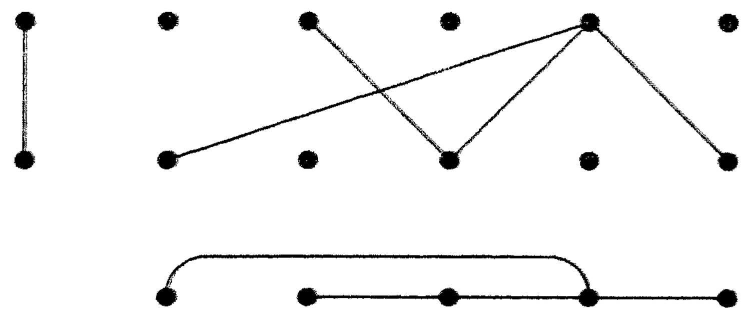

Figure 5.2: A component of $G \times H$ with crossing edges

different components contrary to our assumption. The result follows. ■

Thus if we examine the depiction of $G \times H$ (with the natural ordering on $G$ and $H$), the edges between any two rows in some component of $G \times H$ have the property that no two edges "cross". Observe if $H$ is a tree this may not be the case even with only one colour, as demonstrated in Figure 5.2. Note only one component of $G \times H$ is drawn. Here $H$ is the tree while $G$ is a single edge.

## 5.2 The Path Colouring Algorithm

In this section we assume again that $H$ is an edge-coloured path. We describe a polynomial time algorithm for $H$-COL. We may also assume that $G$ is bipartite and connected. If $G$ is not connected we can apply the algorithm on each component of $G$; if $G$ is not bipartite, then $G \not\to H$. For two edge-colours we know $H$-COL is polynomial by the construction in Chapter Four and the algorithm for oriented paths in [14].

Define a homomorphism $f : G \to G \times H$ to be *G-fixed* if it has the property that for each $g \in G$, $f(g) = (g, h)$ for some $h \in H$. We know from Corollary 1.1.4 that if $G \to H$, then there is a one to one homomorphism, $\phi$, from $G$ into some component $W$ of $G \times H$. Moreover, this homomorphism $\phi$ can always be chosen to be $G$-fixed. Let $W$ be a component of $G \times H$. We denote the set of homomorphisms $G \to W$ by $Hom(G, W)$.

Label the vertices of $H$ with $\{0, 1, 2, ...|V(H)|-1\}$. Let $W$ be a component of $G \times H$. We now define a partial order on the $G$-fixed elements of $Hom(G, W)$. Given a $G$-fixed element $f_1 \in Hom(G, W)$, define $f_1^H$ to be the homomorphism from $G$ to $H$ obtained by composing $f_1$ with the projection map $G \times H \to H$. That is, $f_1(g) = (g, f_1^H(g))$ for all $g \in V(G)$. Let $f_1$ and $f_2$ be two $G$-fixed elements of $Hom(G, W)$. Define $\leq_H$ by $f_1 \leq_H f_2$ if $f_1^H(g) \leq f_2^H(g)$ for all $g \in V(G)$. (Since $H$ is a totally ordered set, this ordering is well-defined.)

**Lemma 5.2.1** *Let $H$ be a fixed edge-coloured path and $G$ a bipartite edge-coloured graph. For each component $W$ of $G \times H$, if the set $Hom(G, W)$ is not empty, then $Hom(G, W)$ contains at least one $G$-fixed homomorphism. Moreover, the set of $G$-fixed homomorphisms in $Hom(G, W)$ has a minimum element with respect to $\leq_H$.*

**Proof.** Let $W$ be a component of $G \times H$ and suppose that $Hom(G, W)$ contains at least one element. By composition with the homomorphism $W \to H$, we conclude $G \to H$. By Corollary 1.1.3, there exists a $G$-fixed homomorphism $\phi : G \to W$. If $\phi$ is the only such element in $Hom(G, W)$, then it is minimum. Suppose $Hom(G, W)$ contains at least two $G$-fixed elements, say $f_1$ and $f_2$. Let $f_3(g) = (g, \min\{f_1^H(g), f_2^H(g)\})$ for all $g \in V(G)$. This minimum is well-defined since $V(H)$ is a totally ordered set.

**Claim** The mapping $f_3 : G \to W$ is a homomorphism.

Suppose $g_1g_2 \in E_i(G)$. The pair $f_3(g_1)f_3(g_2)$ is (by definition of $f_3$) the pair $(g_1, \min\{f_1^H(g_1), f_2^H(g_1)\})\ (g_2, \min\{f_1^H(g_2), f_2^H(g_2)\})$.

Since $f_1$ and $f_2$ are homomorphisms, $f_1(g_1)f_1(g_2)$ and $f_2(g_1)f_2(g_2)$ are each edges in $W$. By Lemma 5.1.3, it must be the case that $(f_1^H(g_1) \leq f_2^H(g_1)$ and $f_1^H(g_2) \leq f_2^H(g_2))$ or $(f_1^H(g_1) \geq f_2^H(g_1)$ and $f_1^H(g_2) \geq f_2^H(g_2))$. In the first case $f_3(g_1) = f_1(g_1)$ and $f_3(g_2) = f_1(g_2)$. In the second case $f_3(g_1) = f_2(g_1)$ and $f_3(g_2) = f_2(g_2)$. Hence, $f_3(g_1)f_3(g_2)$ is an edge in $W$. This establishes the claim.

We conclude that the set of $G$-fixed homomorphisms in $Hom(G, W)$ must have a minimum element. ■

Our aim is to describe an algorithm that finds a minimum $G$-fixed homomorphism from $G$ into a connected component of $G \times H$ and thereby solves $H$-COL in view of Corollary 1.1.4. We have two basic structures. Firstly, $\bar{f}$ is a mapping from $V(G)$ to $V(G \times H)$, which is not necessarily a homomorphism. Secondly, $C$ is a subset of $E_1(G) \cup E_2(G) \cup \cdots \cup E_k(G)$. After choosing a component $W$ of $G \times H$, we have the following two invariants which are true throughout the algorithm.

(i) If $Hom(G, W)$ is not empty, then $\bar{f} \leq_H f$ for all $G$-fixed $f \in Hom(G, W)$.

(ii) For all $\alpha \in \{1, 2, \ldots, k\}$, if $g_1g_2$ is an edge in $E_\alpha(G)\backslash C$, then $\bar{f}(g_1)\bar{f}(g_2)$ is an edge of $E_\alpha(W)$.

We are now ready to describe the Path Colouring Algorithm.

1 Label the components of $G \times H$ with $W_1, W_2, \ldots, W_\omega$.

2 For $m = 1$ to $\omega$ do

2.1 Set $\tilde{f}(g,0) = (g,0)$ for all $g \in V(G)$ and $C = E_1(G) \cup E_2(G) \cup \cdots \cup E_k(G)$ and *valid_map* = *true*.

2.2 While $(C \neq \phi)$ and *valid_map* do

2.2.1 Choose an edge $g_1g_2 \in C$ of colour $\alpha$.

2.2.2 Choose the minimum $(i,j)$ (coordinatewise) such that $(g_1,i)(g_2,j) \in E_\alpha(W_m)$ and $\tilde{f}^H(g_1) \leq i$ and $\tilde{f}^H(g_2) \leq j$. If no such $(i,j)$ exists, then *valid_map* = *false*.

2.2.3 Else (Update the Colouring).

- If $\tilde{f}(g_1) = (g_1,i)$ and $\tilde{f}(g_2) = (g_2,j)$, then continue.
- If $\tilde{f}(g_1) \neq (g_1,i)$ and $\tilde{f}(g_2) = (g_2,j)$, then put all edges incident with $g_1$ into $C$.
- If $\tilde{f}(g_1) = (g_1,i)$ and $\tilde{f}(g_2) \neq (g_2,j)$, then put all edges incident with $g_2$ into $C$.
- If $\tilde{f}(g_1) \neq (g_1,i)$ and $\tilde{f}(g_2) \neq (g_2,j)$, then put all edges incident with $g_1$ and $g_2$ into $C$.

2.2.4 Set $\tilde{f}(g_1) = (g_1,i)$ and $\tilde{f}(g_2) = (g_2,j)$. Remove $g_1g_2$ from $C$.

2.2.5 End While

2.3 If *valid_map* then answer YES and STOP; otherwise next $m$.

3 Answer NO and STOP.

We need to show that the pair $(i,j)$ in step 2.2.2 is well defined. Suppose $(i,j)$ and $(m,n)$ are pairs of vertices in $H$ such that $(g_1,i)(g_2,j) \in E_\alpha(W)$ and $(g_1,m)(g_2,n) \in E_\alpha(W)$, then by Lemma 5.1.3, either $(i,j) \leq (m,n)$ or $(m,n) \leq (i,j)$. Hence, a minimum does exist.

**Theorem 5.2.2** *The Path Colouring algorithm solves H-COL in* $O(|V(G)| + |E(G)|)$ *time when* $H$ *is a fixed path.*

**Proof.** We prove both invariants are true throughout the algorithm by induction on the number of edges checked. Observe this implies that if $C = \phi$, then $\tilde{f}$ is the minimum $G$-fixed homomorphism in $Hom(G, W)$ for some $W$. When zero edges have been checked, both invariants are trivially true. Suppose both are true after $n$ edges have been checked. Further suppose that the $(n+1)^{st}$ edge to be checked is $g_1g_2$. If $Hom(G, W)$ is empty, then invariant (i) it trivially true. If $Hom(G, W)$ is not empty, then let $f$ be the minimum $G$-fixed element of $Hom(G, W)$. We have by induction, $f^H(g_1) \geq \tilde{f}^H(g_1)$ and $f^H(g_2) \geq \tilde{f}^H(g_2)$. Also $f(g_1)f(g_2) \in E_\alpha(W)$ since $f$ is a homomorphism. Therefore, at step 2.2.2 the pair $(i, j)$ exists with $f^H(g_1) \geq i$ and $f^H(g_2) \geq_H j$. The mapping $\tilde{f}$ is updated such that $\tilde{f}(g_1) = (g_1, i)$ and $\tilde{f}(g_2) = (g_2, j)$. By induction, $\tilde{f}(g) \leq_H f(g)$ for all $g \in V(G) \backslash \{g_1, g_2\}$. Therefore, invariant (i) remains true.

Notice we have just proved if $Hom(G, W)$ is not empty, then the pair $(i, j)$ exists at step 2.2.2. Therefore, the algorithm only chooses a new component in $G \times H$ (i.e. Next $m$) if the current $Hom(G, W)$ is empty. In order for the algorithm to reply NO, $Hom(G, W)$ must be empty for all components $W$.

If at step 2.3, a new component is chosen, then returning to step 1 makes both invariants trivially true again. If the pair $(i, j)$ exits in step 2.2.2, then $\tilde{f}$ is updated. By induction, invariant (ii) was true before $\tilde{f}$ was updated. The only edge removed from $C$, and hence the only edge that could make invariant (ii) false, is $g_1g_2$. The choice of $(i, j)$ at step 2.2.2 guarantees that $\tilde{f}(g_1)\tilde{f}(g_2)$ is an edge coloured $\alpha$. Therefore, invariant (ii) remains true. If $C$ becomes empty upon removing $g_1g_2$, then $\tilde{f}$ is a homomorphism and the algorithm has correctly identified a YES instance.

Let $|V(H)| = p$. There are $|V(G)|p$ vertices in $G \times H$. For each edge in $G$ there are at most $2(p-1)$ corresponding edges in $G \times H$. Therefore, $G \times H$ can be constructed in $O(|E(G)| + |V(G)|)$ time. Identifying the components of $G \times H$ requires $O(|E(G)|)$ time. Step 1 requires $O(|E(G)| + |V(G)|)$ time. An edge is added to $C$ when the colour of one of its ends is increased. This means an edge can be added to $C$ at most $2p - 2$ times. Therefore, an edge can be checked at most $2p - 1$ times. Choosing the minimum pair $(i, j)$ in step 2.2.2 requires constant time. Therefore, we require at most $(2p - 1)|E(G)|$ iterations each of constant time. The total time required is $O(|V(G)| + |E(G)|)$. ∎

## 5.3 NP-complete trees

The authors of [14] have constructed NP-complete oriented trees. These trees are large (288 vertices) and complex. Define an *edge-coloured tree* to be an edge-coloured graph whose underlying graph is a tree. Based on a reduction similar to the one in [14], we construct edge-coloured NP-complete trees; however, the use of several edge-colours allows us to construct smaller, simpler trees. In fact, the two trees presented are ***generalized stars***; a ***generalized star*** is a tree with a unique vertex of degree greater than two. Clearly, $H$-COL is in NP. Therefore, we need only provide a polynomial reduction from an NP-complete problem. We use ONE-IN-THREE 3SAT. (See [12] for details on the complexity.) Formally, ONE-IN-THREE 3SAT is defined below.

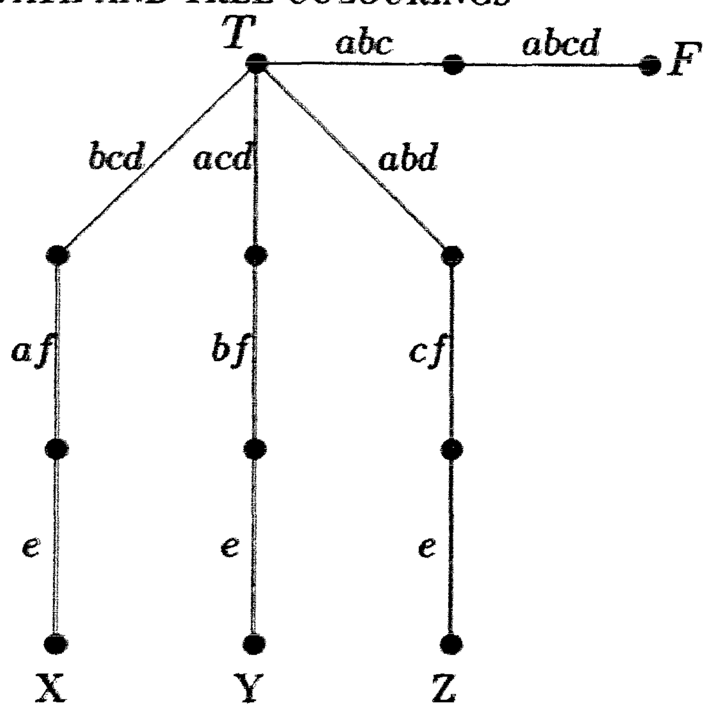


Figure 5.3: The NP-complete generalized star $H$

**ONE-IN-THREE 3SAT**

INSTANCE: Set $\mathcal{U}$ of variables, collection $\mathcal{C}$ of clauses over $\mathcal{U}$ such that each clause $C \in \mathcal{C}$ has $\|C\| = 3$.

QUESTION: Is there a truth assignment for $\mathcal{U}$ such that each clause in $\mathcal{C}$ has exactly one true literal?

This problem remains NP-complete if no $C \in \mathcal{C}$ contains a negated literal.

Let $H$ be the tree in Figure 5.3. The edge-colours are given by the letters beside each edge. For example, if the edge $uv$ has $abc$ beside it, then $u$ is connected to $v$ by edges of colours $a, b,$ and $c$. In other words, under any homomorphism the edges that map to $uv$ may have only the colours $a, b,$ or $c$.

**Theorem 5.3.1** *Let $H$ be the tree in Figure 5.3. Then H-COL is NP-complete.*

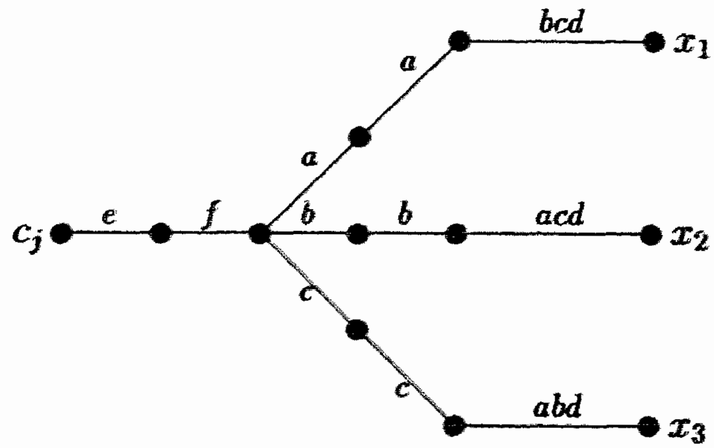


Figure 5.4: The tree $S$.

**Proof.** Given an instance of ONE-IN-THREE 3SAT without negated variables, we construct an edge-coloured graph $G$. Let $S$ be the tree in Figure 5.4; let $G$ have vertices $l_1, l_2, \ldots, l_m$, corresponding to the $m$ literals in our instance of ONE-IN-THREE 3SAT. For each clause, $C_j \in C$ with $C_j = l_{j_1} \vee l_{j_2} \vee l_{j_3}$, we take a copy of $S$ and identify the vertices $x_1, x_2,$ and $x_3$ with $l_{j_1}, l_{j_2},$ and $l_{j_3}$.

Note that any homomorphism $f : S \to H$ maps $c_j$ to $X, Y$ or $Z$. Moreover,

- if $f(c_j) = X$, then $f(x_1) = T$, $f(x_2) = F$ and $f(x_3) = F$,
- if $f(c_j) = Y$, then $f(x_1) = F$, $f(x_2) = T$ and $f(x_3) = F$,
- if $f(c_j) = Z$, then $f(x_1) = F$, $f(x_2) = F$ and $f(x_3) = T$.

Finally, there are homomorphisms that realize each of the three cases above. The verification of these statements is straightforward and is left to the reader.

We shall show that the edge-coloured graph $G$ maps to $H$ if and only if a truth assignment exists that assigns "true" to exactly one variable in each clause $C_j$. Suppose

$f : G \to H$. As observed above, each vertex $c_j$ gets mapped to $X, Y$ or $Z$. Further, once $c_j$ is mapped, the rest of the vertices in $S$ have their images in $H$ uniquely determined. For example, if $f(c_j) = X$, then $f(x_1) = T$ and $f(x_2) = f(x_3) = F$, interpret this as assigning "true" to $l_{j_1}$ and "false" to $l_{j_2}$ and $l_{j_3}$.

On the other hand, given a truth assignment, map all true literals to $T$ and all false literals to $F$. Using the observations above it can be verified that this can be extended to a homomorphism $f : G \to H$. ∎

The above example is nice in that $H$ contains only 12 vertices. The NP-complete directed tree found in [14] has 288 vertices. An example of a two-edge-colour NP-complete tree exists on 98 vertices (see below). It seems that allowing coloured edges lets us observe richer behaviour in smaller examples.

Now we construct an NP-complete tree with two edge-colours. Let $H$ be the tree in Figure 5.5. The labels on the edges here are not colours, but paths found in Figure 5.7. Each path consists of a blue path, followed by a red path, followed by a blue path, followed by a path consisting of a single red edge. The number above each edge in Figure 5.7 corresponds to the length of the path. For example, the path $P_1$ is a path composed of 3 blue edges, 5 red edges, 5 blue edges, and a single red edge. Each path has an orientation from the white vertex on the left to the black vertex on the right. Each label in Figures 5.5 and 5.6 corresponds to an oriented path from Figure 5.7, except for $Q'$ (not shown in Figure 5.7) which is a path of length six whose edges alternate red and blue. That is, $Q'$ is obtained from $Q$ be adding a blue edge then a red edge to the right end of $Q$.

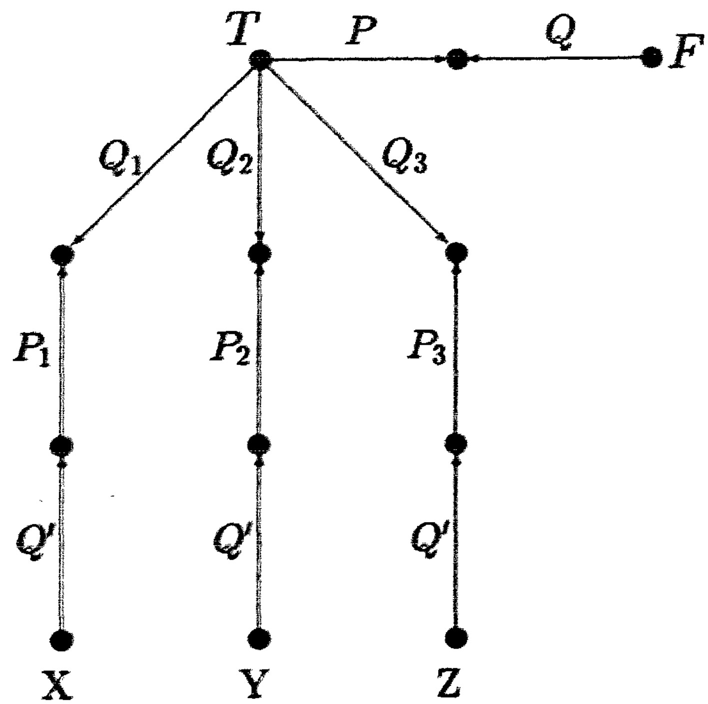


Figure 5.5: The NP-complete two-edge-colour tree $H$.

We now tackle the somewhat cumbersome task of describing homomorphisms between the paths. Consider the following easy proposition.

**Proposition 5.3.2** *Let $W = w_0w_1 \ldots w_{2i+1}$ and $V = v_0v_1 \ldots v_{2j+1}$ be two paths with all edges blue. There exists a homomorphism, $f : W \to V$, such that $f(w_0) = v_0$ and $f(w_{2i+1}) = v_{2j+1}$ if and only if $i \geq j$.*

What does this mean in terms of our paths? Let $W$ and $V$ be two paths from Figure 5.7 (neither of which is $Q'$). Suppose there exists a homomorphism from $W$ to $V$ mapping the white (resp. black) vertex of $W$ to white (resp. black) vertex of $V$. The initial sequence of blue edges in $W$ must map onto the initial sequence of blue edges in $V$ with the ends in $W$ mapping to the corresponding ends in $V$. Also, both sequences have odd length. By Proposition 5.3.2, this can only occur if the sequence in

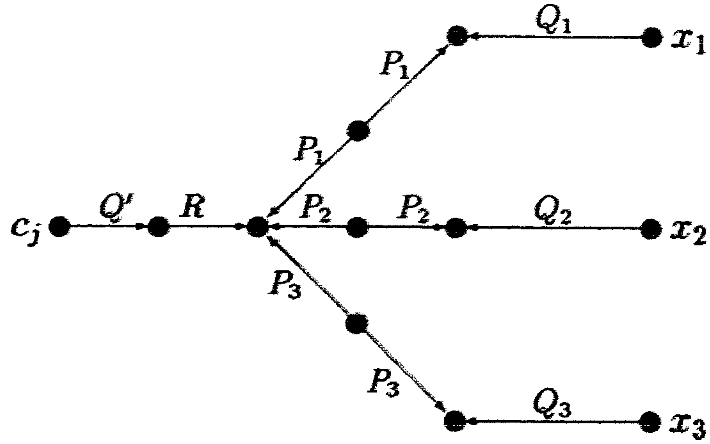


Figure 5.6: Clauses for the Two-edge-coloured Tree

$W$ is at least as long as the sequence in $V$. Now the second monochromatic sequences in $W$ and in $V$ are red. Each sequence has odd length and the ends of $W$ must map to the corresponding ends in $V$. Therefore, the first red sequence in $W$ must be at least as long as the first red sequence in $V$. In other words, $W$ maps to $V$ if and only if each monochromatic sequence in $W$ is at least as long as the corresponding sequence in $V$.

For example, the path $P_1$ will map to the paths $P$, $Q_2$, $Q_3$, and $Q$, but it will not map to the paths $P_2$, $P_3$, $Q_1$, or $R$. The path $Q'$ will only map to an *alternating path* of length six, i.e. a copy of $Q'$. If one checks the tree in Figure 5.5, the only such paths are the three $Q'$ paths incident with $X$, $Y$, and $Z$.

**Theorem 5.3.3** *Let $H$ be the tree in Figure 5.5. Then H-COL is NP-complete.*

**Proof.** The proof works in exactly the same way as for the previous tree. Suppose we are given an instance of ONE-IN-THREE 3SAT without negated variables. We construct a graph $G$ using the tree $S$ in Figure 5.6 for each clause. Because the path

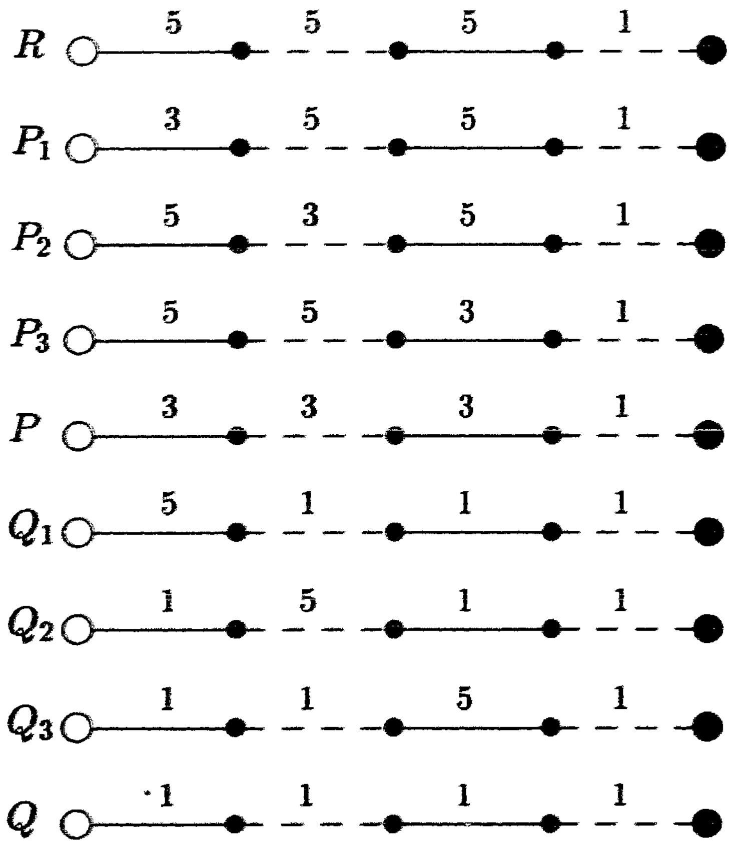


Figure 5.7: Super Edges

$Q'$ in each $C_j$ only maps to one of the paths labeled $Q'$ in $H$, the clauses map to $H$ in the same way as described in Theorem 5.3.1 ∎

Since this chapter has been written, P. Hell, J. Nešetřil, and X. Zhu have discovered new constructions of NP-complete trees. Their trees are much smaller than the trees in [14] but still not as small as our example on 12 vertices.

## 5.4 Characterizing Homomorphisms to Paths

In [20] it is shown that for any oriented graph $G$ and any oriented path $P$, there exists a homomorphism $G \to P$ if and only if all paths homomorphic to $G$ are homomorphic to $P$. In this section we present a similar result for edge-coloured paths. Namely

$G \to P$ if and only if all paths homomorphic to $G$ are homomorphic to $P$ and $G$ contains no odd cycles. Note the second condition (odd cycles) simply states that $G$ must be bipartite as no odd cycle is homomorphic to a path.

The result in [20] does not require odd cycles in the "obstruction set" since one can show that if there is an odd cycle homomorphic to $G$, then the odd cycle must have net length at least one; moreover, this implies there is a path $W$ such that $W \to G$ but $W \not\to P$. Consider the path, $W$, consisting of $k$ "wrappings" of this odd cycle. The path $W$ must have net length at least $k$. By making $k$ sufficiently large, $W$ will have net length longer than $|P|$; hence, $W \not\to P$. For example, suppose $G$ contains a three-cycle with two forward arcs and one backward arc. This cycle has net-length two. The path $W$ consisting of 10 copies of "two-forward-one-backward" has net length 20 and maps to the three-cycle in $G$. If $|P|$ is less than 20, then clearly $W \not\to P$.

In the case of edge-coloured paths we need the explicit condition that $G$ contains no odd cycles, i.e. the existence of an odd cycle in $G$ does not imply the existence of a path $W$ such that $W \to G$ and $W \not\to P$ as it does in the oriented path case. Let $G$ be a blue three-clique and let $P$ be a single blue edge. Any path that maps to $G$ also maps to $P$, yet $G \not\to P$. In other words, paths alone do not suffice as the obstructions for an edge-coloured path $P$.

Conversely, an analogous result for undirected graphs (one edge-colour) is simply that a graph is bipartite if and only if it contains no odd cycles. That is, the obstruction set for an undirected, one edge-colour path is simply the set of odd cycles and one does not need to include paths in the obstruction set. It is easy to see that odd cycles alone do not suffice as the obstruction set for an edge-coloured path. In other words, edge-coloured paths combine some aspects of both undirected paths and oriented paths.

**Theorem 5.4.1** *Let $G$ be an edge-coloured graph and $P$ an edge-coloured path. Then $G \to P$ if and only if for all edge-coloured paths $W$, $W \to G$ implies $W \to P$ and $G$ contains no odd cycles.*

**Proof.** Suppose $g : G \to P$ is a homomorphism. Firstly $G$ contains no odd cycles, since a homomorphism from $G$ to $P$ implies there exists a homomorphism from the underlying graph of $G$ to the underlying graph of $P$. Moreover, the underlying graph of $P$ is bipartite which implies the underlying graph of $G$ is bipartite since a nonbipartite graph cannot be homomorphic to a bipartite graph.

Let $W$ be a path such that $f : W \to G$ is a homomorphism. By composition with $g$ we have $g \circ f : W \to P$ is a homomorphism. This proves the necessity of the condition.

On the other hand, suppose $G$ contains no odd cycles and for all paths $W$, $W \to G$ implies $W \to P$. We will prove that $G \to P$. We shall in fact prove a stronger statement. Let $H$ be a two-clique, on vertices $\{0, 1\}$, containing an edge of each colour occurring in either $G$ or $P$. Then since both $G$ and $P$ are bipartite, there exists homomorphisms $c_G : G \to H$ and $c_P : P \to H$. We shall show that we can choose $c_G$ and $c_P$ so that if for all paths $W$, $W \to G$ implies $W \to P$, then there exists a homomorphism $g : G \to P$ such that $c_P \circ g = c_G$. This stronger statement implies the sufficiency of Theorem 5.4.1.

**Lemma 5.4.2** *Let $G$, $P$, and $H$ be as above. There exist homomorphisms $c_G : G \to H$ and $c_P : P \to H$ such that given a path $W$, for all homomorphisms $f : W \to G$ there exists a homomorphism $f' : W \to P$ such that $c_G \circ f = c_P \circ f'$.*

**Proof** Assume that $G$ is connected, (otherwise, treat each component of $G$ separately). Let $c_G : G \to H$ be a homomorphism. This homomorphism induces a partition of $G$ into two independent sets $G_0$ and $G_1$. Similarly, the homomorphism $c_P : P \to H$ induces a partition of $P$ into independent sets $P_0$ and $P_1$.

Suppose to the contrary that the lemma does not hold; that is, there do not exist $c_G$ and $c_P$ with the composition property. Then without loss of generality there exists a path $W_v$ starting at $v$ and a homomorphism $f_1 : W_v \to G$ such that $f_1(v) \in G_0$ so that all homomorphisms $f_1' : W_v \to P$ have the property that $f_1'(v) \in P_0$ and there exists a path $W_u$ starting at $u$ and a homomorphism $f_2 : W_u \to G$ so that $f_2(u) \in G_0$ and all homomorphisms $f_2' : W_u \to P$ have the property that $f_2'(u) \in P_1$.

Let $W_{uv}$ be a path from $f_2(u)$ to $f_1(v)$ in $G$. Notice, $W_{uv}$ is a path in $G$ and hence $W_{uv} \to G$. Let $W$ be the path formed by identifying $v \in W_v$ and $f_1(v) \in W_{uv}$ and identifying $u \in W_u$ and $f_2(u) \in W_{uv}$. That is, identify corresponding ends of the paths. Since $W_v$, $W_u$, and $W_{uv}$ admit homomorphisms $f_1$, $f_2$, and $id$ (indentity) to $G$, we have $W \to G$. Moreover, by the assumptions in the statement of the lemma, this implies there exists a homomorphism $f' : W \to P$. The path $W_{uv}$ must have even length since $f_1(v) \in G_0$ and $f_2(u) \in G_0$. However, $f'(v) \in P_0$ and $f'(u) \in P_1$ since both $W_v$ and $W_u$ are subpaths of $W$ and we are assuming that all homomorphisms of $W_v$ (resp. $W_u$) map $v$ (resp. $u$) to a vertex in $P_0$ (resp. $P_1$). Hence, the image of $W_{uv}$ in $P$ is an odd length path. This is impossible. The result follows. ∎

We now prove our stronger statement.

**Theorem 5.4.3** *Let $G$ be a bipartite edge-coloured graph and $P$ an edge-coloured path and let $c_G$ and $c_P$ be the homomorphisms defined above. Suppose for all paths $W$, $W \to G$ implies $W \to P$. Then there exists a homomorphism $g : G \to P$ such that $c_P \circ g = c_G$.*

**Proof** We now describe the homomorphism $g : G \to P$. Given a path $W$ beginning at a vertex $v$, denote this by $b(W) = v$. In the following definitions we use the notation $f : G \to H$ to indicate there exists a homomorphism $f : G \to H$. Define:

$$\begin{aligned}
\phi_0(W) &= \max\{v \in P \| f' : W \to P, f'(b(W)) = v, c_P(v) = 0\} \\
\phi_1(W) &= \max\{v \in P \| f' : W \to P, f'(b(W)) = v, c_P(v) = 1\}
\end{aligned}$$

Note: Normally one defines the maximum of the empty set to be zero. However, for our purposes we say $\phi_0$ is undefined if the maximum is taken over the empty set. In the case $\phi_0(W)$ is undefined, all homomorphisms $W \to P$ map $b(W)$ to a vertex in $P_1$. A similar note applys to $\phi_1$.

Define:

$$\psi(v) = \min\{\phi_i(W) : v \in G_0, f : W \to G, f(b(W)) = v\} \text{ for } v \in G_i$$

Observe that each of $\psi$ is well-defined since Lemma 5.4.2 implies the minimums above are taken over nonempty sets.

We now prove that $\psi$ is the desired map. That is, $\psi$ is a homomorphism such that $c_P \circ W = c_G$.

**Claim 5.4.3.1** *Let $uv \in E_i(G)$. Then $\psi(u) \neq \psi(v)$.*

**Proof of Claim** Let $c_G(v) = 1$. The value of $\psi(v)$ is a minimum over a subset of the vertices of $P$ coloured 1. The value of $\psi(u)$ is a minimum over a subset of the vertices of $P$ coloured 0. These two sets are disjoint. ∎

**Claim 5.4.3.2** *The mapping $\psi$ is a homomorphism.*

**Proof of Claim.** Let $uv$ be an edge in $G$. Without loss of generality assume $uv$ is blue and $c_G(u) = 0$. Let $W_u$ be a path such that $\psi(u) = \phi_0(W_u)$. Let $B$ be a single blue edge. The path $B \circ W_u$ is the path formed by identifying the end of $B$ with the beginning of $W_u$. The path $B \circ W_u$ is a path that maps to $G$ so that $f(b(B \circ W_u)) = v$. Therefore, $\phi_1(B \circ W_u)$ is defined. Observe that any homomorphism mapping $B \circ W_u$ to $P$ can not map its start to a value larger than $\phi_0(W_u)+1$; otherwise, this homomorphism restricted to $W_u$ would map $b(W_u)$ to a value larger than $\phi_0(W_u)$. Also observe that $\psi(v)$ must be no larger than $\phi_1(B \circ W_u)$ by definition of $\psi$. Hence, we have:

$$\psi(v) \leq \phi_1(B \circ W_u) \leq \phi_0(W_u) + 1 = \psi(u) + 1$$

Similarly,

$$\psi(u) - 1 \leq \psi(v)$$

Recall $\psi(u) \neq \psi(v)$. Therefore, either $\psi(v) = \psi(u) + 1$ or $\psi(v) = \psi(u) - 1$. Assume $\psi(v) = \psi(u) + 1$. (The other case is similar.)

Let $h : B \circ W_u \to P$ be the homomorphism that defines $\phi_1(B \circ W_u)$. By restricting $h$ to $W_u$ we see that $h(u) \leq \phi_0(W_u) = \psi(u)$. Also, $h(v)$ must be larger than $\psi(v)$ since $\psi$ is a minimum. We have

$$h(v) \geq \psi(v) = \psi(u) + 1 > \psi(u) \geq h(u)$$

Also note, $\|h(u) - h(v)\| = 1$. Therefore, $h(u) = h(v) - 1$. By the above inequalities we see $\psi(v) = h(v)$ and $\psi(u) = h(u)$. Since $h$ is a homomorphism, there must be a blue edge between $h(u)$ and $h(v)$. Hence $\psi(v)\psi(u)$ is a blue edge in $P$. Thus, $\psi$ is a homomorphism. ∎

The observation required to complete the proof Theorem 5.4.3 is that if $v \in G_0$, then $\psi(v) \in P_0$. Hence, $c_P \circ \psi = c_G$. ∎

The existence of $\psi$ completes the proof of the necessity of the condition in Theorem 5.4.1.

The proof above that $\psi$ is a homomorphism can be extended to the case when $G$ and $P$ have directed, coloured edges. Hence, the following stronger result is true:

**Corollary 5.4.4** *Let $G$ be an edge-coloured, directed graph and $P$ be an edge-coloured path with directed edges. Then $G \to P$ if and only if $G$ contains no odd cycles and for all paths $W$, $W \to G$ implies $W \to P$.*

# Chapter 6

# Cycles

In this chapter we study $H$-COL where $H$ is a digon-free edge-coloured cycle. The digon-free restriction is assumed for the remainder of this chapter. The emphasis is on two edge-colours, although some results naturally generalize to more edge-colours. An edge-coloured cycle can be viewed as being composed of monochromatic paths. A maximal monochromatic path is called a *piece*. For example consider a cycle of length eight with the first three edges red, the next edge blue, the next two edges red, and the final two edges blue. This cycle has four pieces; two red and two blue. The red pieces have length three and two. The blue pieces have length one and length two. See Figure 6.1.

Let $H$ be a two-edge-coloured cycle. We characterize the complexity of $H$-COL by the number and the parity of the pieces in $H$. We show that $H$-COL is polynomial if all the pieces of $H$ have odd length. In fact, this result generalizes to $k \geq 2$ edge-colours. If $H$ contains exactly one even length piece, then $H$-COL is polynomial. We show that if $H$ is a two-edge-coloured cycle with all pieces having even length, then $H$-COL is polynomial. These results imply that any cycle consisting of two pieces is polynomial.

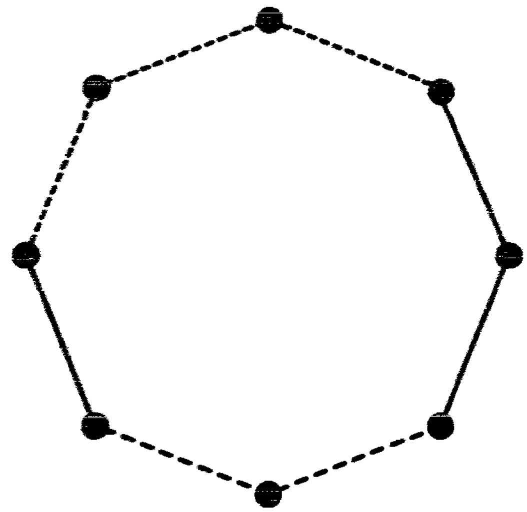

Figure 6.1: An edge-coloured cycle with four pieces.

Since we restrict our attention to two-edge-coloured cycles, the number of pieces must be even. Using the results stated above plus an *ad hoc* algorithm we show for any two-edge-coloured cycle, $H$, with four pieces, $H$-COL is polynomial. On the other hand, consider a two-edge-coloured cycle, $H$, where each red piece has odd length and each blue piece has even length. We show that $H$-COL is NP-complete if $H$ has six pieces. In fact we show that for such a cycle $H$ with $k \geq 4$ pieces, $H$-COL is polynomial if $k \equiv 0 \pmod 4$ and $H$-COL is NP-complete if $k \equiv 2 \pmod 4$.

Before we present the results we make one final observation. Let $G$ be an instance of some $C$-colouring problem, where $C$ is an edge-coloured cycle. Many of the algorithms presented here begin by defining a mapping from the mixed vertices of $G$ to $C$. In the case that $G$ does not contain any mixed vertices, if $G$ admits a homomorphism to $C$, then $G$ must map to a single piece of $C$. Hence, if $G$ contains only edges of colour $i$, then $G \to C$ if and only if $G$ is bipartite and $C$ contains at least one piece of colour $i$. Therefore, in the following assume $G$ has at least one mixed vertex.

## 6.1 The Mixed Vertex Homomorphism Problem

Let $H$ be an edge-coloured cycle and let $G$ be an instance of $H$-COL. In many of the algorithms below we use the following strategy:

- define a function, $f$, from the mixed vertices of $G$ to the mixed vertices of $H$,
- extend this function to a homomorphism from $G$ to $H$.

We begin by examining the complexity of extending $f$ to a homomorphism. In fact, we shall show that when $H$ is an edge-coloured cycle, this "extension" problem is polynomial. To this end we define a problem that one might consider a partial HFP problem and then we examine this problem for the specific case that $H$ is an edge-coloured cycle. Formally:

Let $H$ and $Y$ be fixed edge-coloured graphs and let $h : H \to Y$ be a homomorphism.

**Mixed Vertex Homomorphism Problem** – $MVHP(H, h, Y)$

INSTANCE: An edge-coloured graph $G$ and a homomorphism $g : G \to Y$ .
QUESTION: Does there exist a homomorphism $f : G \to H$ such that for all mixed vertices $v \in V(G)$, $h \circ f(v) = g(v)$?

Let $H$ and $Y$ be edge-coloured graphs and $h : H \to Y$ a homomorphism. We show that $G$ is a YES instance of $H$-COL if and only if there exists $g : G \to Y$, such that $G, g$ is a YES instance of $MVHP(H, h, Y)$. Suppose $G$ is a YES instance of $H$-COL, then there exists $f : G \to H$. Clearly, $G, h \circ f$ is a YES instance of $MVHP(H, h, Y)$. On the other hand, suppose $G, g$ is a YES instance of $MVHP(H, h, Y)$, then there exists an $f : G \to H$ which implies $G$ is a YES instance of $H$-COL.

Suppose $C$ is an edge-coloured cycle with pieces $P_0, P_1, P_2, \ldots, P_k$. Label the mixed vertices of $C$ with $m_0, m_1, \ldots, m_k$ so that $m_i$ is the vertex common to pieces $P_{i-1}$ and $P_i$. Let $m_0$ be the mixed vertex common to pieces $P_k$ and $P_0$.

We construct a new edge-coloured cycle, $C'$, with vertex-set $\{c'_0, c'_1, \ldots, c'_k\}$. For each piece, $P_i$, in $C$ add the following to $C'$. If $P_i$ has odd length and is of colour $t$, then add the edge $c'_i c'_{i+1}$ to $E_t(C')$. If $P_i$ has even length and is of colour $t$, then add a vertex $c'_{i+\frac{1}{2}}$ to $C'$ and put a path of colour $t$ on the vertices $c'_i c'_{i+\frac{1}{2}} c'_{i+1}$ in $C'$.

For an example see Figure 6.2. The cycle $C$ consists of four pieces. The piece $P_0$ is blue and has length four. The piece $P_1$ is red and has length two, etc. The labels of the pieces are outside the cycle. The labels of the mixed vertices are inside the cycle. The cycle $C'$ consists of four pieces as well; one piece for each piece in $C$. The piece replacing $P_0$ is on vertices $\{c'_0, c'_{0+\frac{1}{2}}, c'_1\}$, etc.

We claim there exists a homomorphism $h : C \to C'$. Define $h$ initially from the mixed vertices of $C$ to the mixed vertices of $C'$ as follows:

$$h(m_i) = c'_i \quad \text{for} \quad 0 \leq i \leq k$$

It is straightforward to check that $h$ extends to a homomorphism from $C$ to $C'$. We are now ready to prove the following theorem.

**Theorem 6.1.1** *Let $C$ be an edge-coloured cycle. Let $C'$ and $h$ be defined as above. Then $MVHP(C, h, C')$ is polynomial.*

**Proof.** Let $G, g$ be an instance of $MVHP(C, h, C')$. We begin by defining a function, $f$, from the mixed vertices of $G$ to the mixed vertices of $C$. For each mixed

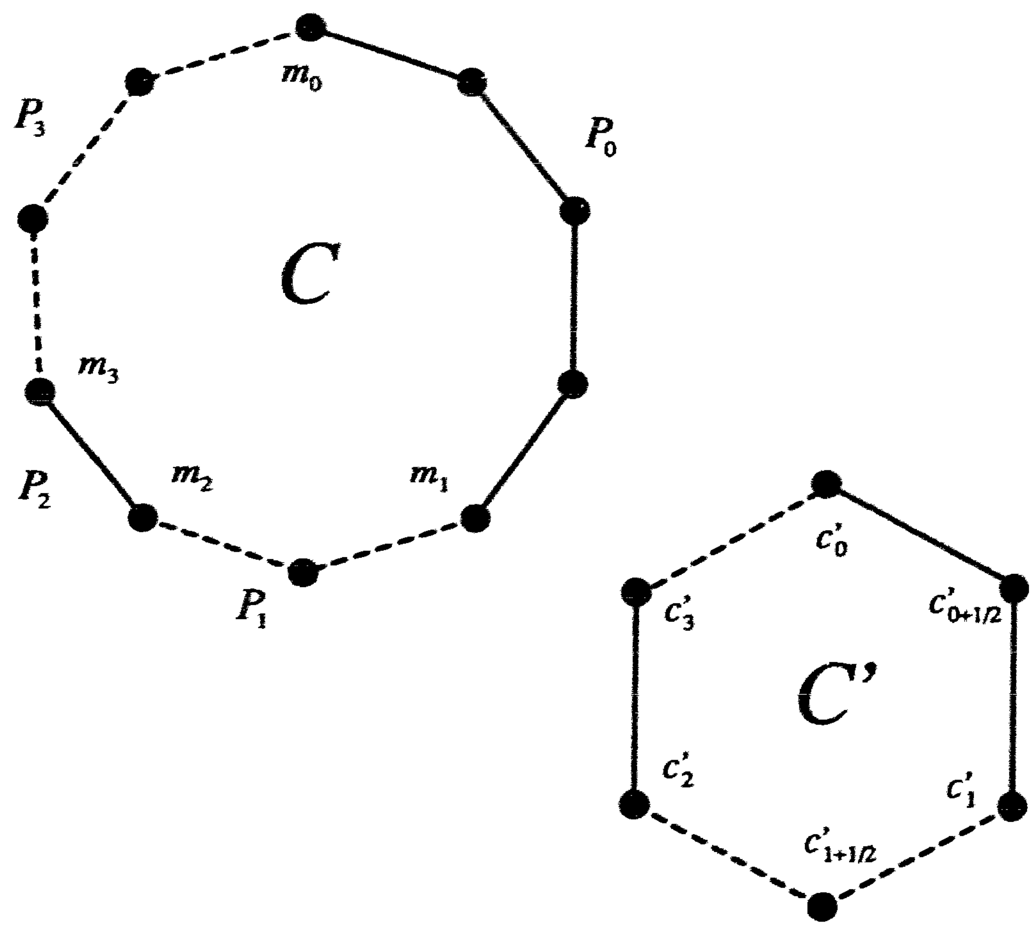


Figure 6.2: The construction of $C'$ from $C$.

vertex $v$ in $G$ define:

$$f(v) = m_i \quad \text{if and only if} \quad g(v) = c_i'.$$

It is easy to see that if there exists a homomorphism $\tilde{f} : G \to C$ such that $h \circ \tilde{f}(v) = g(v)$ for all mixed vertices $v$ in $G$, then $\tilde{f}(v) = f(v)$ for all mixed vertices $v$ in $G$. To see this observe that the only mixed vertex in $C$ that is mapped to $c_i'$ by $h$ is $m_i$.

Suppose $S_i$ is the set of mixed vertices in $G$ that is mapped to $m_i$ by $f$ for $0 \le i \le k$. Consider $S_i$ and $S_{i+1}$. Suppose that piece $P_i$ is colour $t$. Let $d_i$ be the minimum distance in colour $t$ from a vertex in $S_i$ to a vertex in $S_{i+1}$. Define $d_i$ for $0 \le i \le k-1$. Similarly, define $d_k$ using $S_0$ and $S_k$. Note that $d_i$ can be infinite.

Denote the length of piece $P_i$ by $\|P_i\|$. We claim that $f$ can be extended to a homomorphism if and only if $d_i \ge \|P_i\|$ for $0 \le i \le k$. The necessity of the condition is obvious. Suppose $u$ is a vertex, that is a not a mixed vertex, of $G$. Assume $u$ is incident with only edges of colour $t$. We explain how to extend $f$ to $u$. Let $P_i$ be a piece of colour $t$. Suppose there are paths, $Q_1$ and $Q_2$, from $u$ to a mixed vertex in $S_i$ and to a mixed vertex in $S_{i+1}$ respectively, which contain no mixed vertices except for the end of the paths. Then $g$ must map $u$ to one of $\{c_i', c_{i+\frac{1}{2}}', c_{i+1}'\}$. Hence, any path from $u$ to a mixed vertex (which contains no mixed vertices except for the end of the path) must terminate in either $S_i$ or $S_{i+1}$. It must be the case that the sum of the lengths of these two paths is at least $\|P_i\|$ (and the two paths must have the same parity). If $\|Q_1\|$ is less than or equal to $\lfloor \frac{\|P_i\|}{2} \rfloor$, then map $u$ to a vertex in $C$ distance $\|Q_1\|$ from $m_i$. Similarly if $\|Q_2\|$ is less than half the length of $P_i$, then map $u$ to a vertex distance $\|Q_2\|$ from $m_{i+1}$. If both $\|Q_1\|$ and $\|Q_2\|$ are greater than $\lfloor \frac{\|P_i\|}{2} \rfloor$, then map $u$ to either $\lfloor \frac{\|P_i\|}{2} \rfloor$ or $\lceil \frac{\|P_i\|}{2} \rceil$ depending on the parity of $\|Q_1\|$. ■

## 6.2 All Pieces Have The Same Parity

We now examine edge-coloured cycles where each piece has the same parity. We begin by examining two-edge-coloured cycles with all pieces having even length.

**Lemma 6.2.1** *Let $C$ be a two-edge-coloured cycle such that each piece has even length. Then $C$-COL is polynomial.*

**Proof.** Suppose the edge-colours of $C$ are red and blue. We claim $C$ retracts to an edge-coloured path of length two. Let $P$ be the path on $\{-1, 0, 1\}$ where $-10$ is red and $01$ is blue. Firstly, observe that for a given vertex $v \in C$ all paths from $v$ to mixed vertices have the same parity, since every piece in $C$ is even. Define $f : C \to P$ as follows: $f(v) = -1$ if $v$ is a red only vertex and there exists an odd length path from $v$ to a mixed vertex; $f(v) = 1$ if $v$ is blue only and there exists an odd length path from $v$ to a mixed vertex; and $f(v) = 0$ otherwise. It is easy to see that $f$ is a homomorphism. Moreover, $C$ contains a copy of $P$ and $f$ is the identity map on this copy of $P$. Thus, $C$ retracts to $P$. By the Forcing Lemma (Lemma 2.2.3) or by the results on paths, $P$-COL is polynomial and hence $C$-COL is polynomial. ∎

Most of this chapter focuses on two-edge-coloured cycles; however, the following lemma in fact holds for cycles with at least two edge-colours.

**Lemma 6.2.2** *Let $C$ be an edge-coloured cycle, on at least two edge-colours, where each piece has odd length. Then $C$-COL is polynomial.*

**Proof.** Let $C'$ and $h$ be defined as in Theorem 6.1.1. Let $G$ be an instance of $C$-COL. We can assume $G$ is connected, otherwise we treat each component of $G$ separately. Since each piece in $C$ has odd length, each piece in $C'$ is a single edge.

The cycle $C'$ satisfies the conditions of the Forcing Lemma (Lemma 2.2.3) and $G$ is connected; therefore, we can construct all homomorphisms of $G$ to $C'$ in polynomial time. As observed above, $G \to C$ if and only if there exists a homomorphism $g : G \to C'$ such that $G, g$ is a YES instance of $MVHP(C, h, C')$. ∎

The above two lemmas give us the following theorem.

**Theorem 6.2.3** *Let $C$ be a two-edge-coloured cycle. Suppose each piece has the same parity. Then $C$-COL is polynomial.*

## 6.3 One Even Piece

Through use of the MVHP we show in this section that for any two-edge-coloured cycle, $C$, with exactly one even length piece, $C$-COL is polynomial.

**Theorem 6.3.1** *Let $C$ be a two-edge-coloured cycle with exactly one even length piece. Then $C$-COL is polynomial.*

**Proof.** Let $P_0, P_1, \ldots, P_k$ be the pieces of $C$ such that $P_k$ is the unique even length piece. Using the convention above, $m_k$ and $m_0$ are the mixed vertices at either end of $P_k$. Assume that the edge-colours are red and blue and that $P_k$ is red.

Construct $C'$ and $h$ as above. The edge-coloured cycle $C'$ has the property that every piece is a single edge with the exception of the red path of length two: $c'_k c'_{k+\frac{1}{2}} c'_0$. This implies that every vertex $c'_i$ in $C'$, with the exception of $c'_{k+\frac{1}{2}}$, is incident with one blue edge and one red edge. Hence, using ideas from the Forcing Lemma (Lemma 2.2.3), if vertex $v \in V(G)$ is mapped to $c'_i \in V(C')$, then the image of some set of vertices in $G$, say $X$, is uniquely determined. This is discussed in more detail below.

Our strategy is as follows: Let $G$ be an instance of $C$-COL. We construct a homomorphism $g : G \to C'$ so that $G, g$ is a YES instance of $MVHP(C, h, C')$ or conclude no such $g$ exists and hence $G$ is a NO instance of $C$-COL.

Our first observation concerns pairs of mixed vertices, in $G$, joined by an even length red path. Let $u$ and $v$ be two such vertices. We observe if $G$ admits a homomorphism to $C$ the following is true: If $u$ maps to any mixed vertex other than $m_0$ or $m_k$, then $v$ must map to the same mixed vertex. This follows from the fact that $P_k$ is the only even length piece. Also, if $u$ maps to $m_0$ or $m_k$ and the length of the shortest red path from $u$ to $v$ is less that $\|P_k\|$, then $u$ and $v$ map to the same vertex in $C$. Hence, we begin by preprocessing $G$ in the following way: Define an equivalence relation on the mixed vertices of $G$. Initially, for a pair of mixed vertices $u$ and $v$, we say $u$ and $v$ are related if $u$ and $v$ are joined by a red path of length less than $\|P_k\|$. The equivalence relation is defined by taking the transitive closure of this initial relation. Now, for each equivalence class $X$, identify all mixed vertices belonging to $X$. This preprocessing ensures that any homomorphism $g : G \to C'$ has the property for any mixed vertices $u$ and $v$ in $G$ such that $g(u) = c_0'$ and $g(v) = c_k'$, the distance from $u$ to $v$ in red is at least $\|P_k\|$.

We now describe the construction of $g$. Let $u$ be a mixed vertex in $G$. We set $g(u) = c_0'$ and attempt to extend $g$ to a homomorphism from $G$ to $C'$ such that $G, g$ is a YES instance of $MVHP(C, h, C')$. If this is not possible, then we choose a different mixed vertex in $G$, say $w$, and set $g(w) = c_0'$. If testing whether or not $g$ can be extended to a homomorphism requires polynomial time, then attempting to extend $g$ once for each mixed vertex in $V(G)$ requires polynomial time. Hence, we can assume without loss of generality that $u$ is a mixed vertex in $G$ and if $G$ admits a homomorphism, $g$, to $C'$ such that $G, g$ is a YES instance of $MVHP(C, h, C')$, then $g(u) = c_0'$. In other words, the choice to map $u$ to $c_0'$ is correct.

As remarked above, $g(u) = c_0'$ uniquely determines the image in $C'$ of some set of vertices, $X \subseteq V(G)$. Specifically, once $u$ is mapped to $c_0'$, all vertices of an even distance in colour blue from $u$ must be mapped to $c_0'$. All vertices of an odd distance in colour blue from $u$ must be mapped to $c_1'$. That is, the image of these vertices is uniquely determined. Now consider, for example, some vertex $v$ that is mapped to $c_1'$ by $g$. Any vertex of an even distance in red from $v$ must be mapped to $c_1'$. Any vertex of an odd distance in red from $v$ must be mapped to $c_2'$. Similarly, any vertex of an even distance in blue from $v$ must map to $c_1'$ and any vertex of an odd distance in blue from $v$ must map to $c_0'$. Continuing we will determine the image (under $g$) of some set of vertices in $G$. Call this set $X$. Note that the reason this "Forcing-type" argument does not uniquely determine the image of all vertices in $G$ is that a vertex that is an even distance in red from $u$ can map either to $c_0'$ or to $c_k'$.

Let $T$ be the set of red only vertices that are joined by a red path to a vertex mapped to either $c_0'$ or $c_k'$. Let the subgraph induced by $X \cup T$ be denoted $X'$. The map $g$ can be extended to a homomorphism $X' \to C'$ if and only if all red paths from mixed vertices mapped to $c_0'$ to mixed vertices mapped to $c_k'$ have even length. If $g$ can not be extended to a homomorphism then $G$ is not a YES instance of $C$-COL. Hence, assume that it can be extended. Test if $X', g$ is a YES instance of $MVHP(C, h, C')$. If the answer is NO, then $G$ is a NO instance of $C$-COL. Recall that we are assuming the decision $g(u) = c_0'$ is correct and all other images are uniquely determined.

Let $S$ be the set of mixed vertices in $X$ that are mapped to either $c'_0$ or $c'_k$. Let $v$ be a mixed vertex of $G\backslash X$ that is joined by a red path to a vertex in $S$, such that the path has mixed vertices only at its ends. If no such vertex exists, then the set $X$ is all of $V(G)$ and we conclude $G$ is a YES instance of $C$-COL. Therefore, assume such a $v$ exists. The vertex $v$ must map under $g$ to either $c'_0$ or $c'_k$.

Begin by setting $g(v) = c'_0$. As above this uniquely determines the image of some set of vertices under $g$ in $C'$. Add these vertices to the set $X \cup T$. As above let $X'$ be the subgraph induced by $X \cup T$. Test if $X', g$ is a YES instance of $MVHP(C, h, C')$. If the answer is NO, then perhaps the choice to map $v$ to $c'_0$ was wrong. Hence, set $g(v) = c'_k$; add the corresponding vertices to $X \cup T$, and test if $X', g$ is a YES instance of $MVHP(C, h, C')$. If the answer is NO, then $G$ must be a NO instance of $C$-COL. At this point either we answer NO, or we have $X', g$ is a YES instance of $MVHP(C, h, C')$.

We repeat the above process, by choosing a new $v$, and either stop because $G$ is a NO instance of $C$-COL or there is no such $v$ and $G$ is a YES instance of $C$-COL.

To observe that this process is polynomial, note that once $v$ is mapped to $c'_0$ or $c'_k$ and $X', g$ is a YES instance of $MVHP(C, h, C')$ we never need to change the image of $v$. Recall that our strategy is to extend the map $g$ to a homomorphism of $G$ to $C'$. Suppose that $v$ is mapped to $c'_0$ and $w$ is a mixed vertex joined by a red path to a mixed vertex added to $S$ as a result of mapping $v$ to $c'_0$. Further suppose that $g$ can not be extended to a homomorphism if $g(w) = c'_0$ nor can $g$ be extended to a homomorphism if $g(w) = c'_k$. At this point we can STOP and answer NO. It is not the case that mapping $v$ to $c'_k$ will now allow $g$ to be extended. One can see that mapping

$v$ to $c'_k$ will cause the same set of mixed vertices to be added to $S$ as mapping $v$ to $c'_0$. Hence, $w$ will still be joined by a red path to a mixed in $S$ and at some point we must attempt to extend $g$ with $g(w) = c'_0$ or $g(w) = c'_k$.■

## 6.4 Even Blue and Odd Red Pieces

In the following we examine cycles where each blue piece has even length and each red piece has odd length. We show that for a cycle, $C$, with $k$ red pieces (and hence $k$ blue pieces), $C$-COL is polynomial if $k$ is even and $C$-COL is NP-complete if $k$ is odd.

**Theorem 6.4.1** *Let $k \geq 2$ be an integer. Let $C$ be a two-edge-coloured cycle with red and blue edges. Suppose that each blue piece has even length and each red piece as odd length and suppose $C$ has $k$ red pieces. If $k$ is even, then $C$-COL is polynomial. If $k$ is odd, then $C$-COL is NP-complete.*

**Proof.** Suppose $C$ has an even number of red pieces. To see that $C$ retracts to a path, let $P_r$ be a shortest red piece in $C$. Let $P$ be the path formed by adding a single blue edge to either end of $P_r$. It is easy to verify that $C$ retracts to $P$. Hence, $C$-COL is polynomial by the results on paths.

Suppose $C$ contains an odd number of red pieces. We shall use the indicator construction to construct a graph, $C^*$, (multiplicity one) containing no loops and an odd cycle. Since odd cycles yield NP-complete colouring problems, by [19], $C^*$-COL is NP-complete. This implies $C$-COL is NP-complete.

Suppose the longest blue piece in $C$ has length $b$ and suppose the longest red piece in $C$ has length $r$. Let $I$ be a path consisting of a blue path of length $b-1$ followed by a red path of length $r$ followed by a blue path of length $b-1$ and let $i$ and $j$ be the end points of the path $I$. Note that each of the paths comprising $I$ have odd length. Let $C^*$ be the result of the indicator construction with respect of $(I, i, j)$.

Observe that since $I$ has odd length and only one red piece and $k/geq2$, any image of $I$ in $C$ must be a path with $i$ and $j$ mapping to different vertices. Hence $C^*$ is loop-free.

Let $P_0, P_1, P_2$ be three consecutive pieces in $C$, where $P_0$ and $P_2$ are blue and $P_1$ is red. Let $m_1$ be the mixed vertex shared by $P_0$ and $P_1$. Let $m_2$ be the mixed vertex shared $P_1$ and $P_2$. Let $m_3$ be the other mixed vertex in $P_2$, i.e., the other end of $P_2$. Furthermore, let $b_1$ be the blue only vertex of $P_0$ adjacent to $m_1$ and let $b_2$ be the blue only vertex of $P_2$ adjacent to $m_3$. By the lengths of the paths in $I$, one can check that there is a homomorphism of $I$ to $C$ such that $i$ maps to $b_1$ and $j$ maps to $b_2$. Hence the edge $b_1b_2$ is in $C^*$. Since there are an odd number of red pieces in $C$, there are an odd number of such edges. Moreover, these edges form an odd length cycle. Hence, by [19] $C^*$-COL is NP-complete. ∎

## 6.5 Two or Four Pieces.

In this section we examine those two-edge-coloured cycles consisting of exactly two pieces or exactly four pieces. We show that for all such cycles $C$, the complexity of $C$-COL is polynomial.

**Theorem 6.5.1** *Let $C$ be a two-edge-coloured cycle with two pieces. Then $C$-COL is polynomial.*

**Proof** Let $C$ be a cycle with two pieces. In light of Theorem 6.2.3 if both pieces have the same parity, then $C$-COL is polynomial. On the other hand, if one piece is even and one piece is odd, then by Theorem 6.3.1, $C$-COL is polynomial. ∎

Let $C$ be a cycle with four pieces on edge-colours red and blue. In many cases the complexity of $C$-COL follows from previous results. These are summarized below:

**Proposition 6.5.2** *Suppose $C$ is a two-edge-coloured cycle with four pieces of which exactly zero, one or four are even. Then $C$-COL is polynomial.*

**Proof** If all pieces in $C$ have the same parity, i.e. $C$ has zero or four even length pieces, then $C$-COL is polynomial by Theorem 6.2.3. If $C$ has one even length piece and three odd length pieces, then $C$-COL is polynomial by Theorem 6.3.1. ∎

We now consider cycles with four pieces of which two or three of the pieces have even length.

**Proposition 6.5.3** *Let $C$ be a two-edge-coloured cycle with four pieces of which two have even length and are blue and two have odd length and are red. Then $C$-COL is polynomial.*

**Proof** This follows from Theorem 6.4.1. ∎

This leaves two possible configurations. One is a cycle with two of the four pieces adjacent and of even length. The other is a cycle with three of the four pieces even length. The following two theorems complete the classification of cycles with four pieces.

**Theorem 6.5.4** *Let $C$ be a two-edge-coloured cycle with four pieces. Suppose $C$ has exactly two even length pieces and they are adjacent (i.e. they share a vertex). Then $C$-COL is polynomial.*

**Proof** Let the pieces of $C$ be $R_1$, $R_2$, $B_1$, $B_2$, where $R_1$ is an odd length, red path; $R_2$ is an even length, red path; $B_1$ is an odd length, blue path; $B_2$ is an even length, blue path; and a clockwise traversing of the cycle will traverse the pieces in the order $B_1$, $R_2$, $B_2$, $R_1$. Let $v$ be the vertex common to $R_1$ and $B_1$.

Since $C$ has even length, it is bipartite. Partition $C$ into two independent sets, $(C_0, C_1)$, with $v \in C_0$. Using the ideas from Chapter 4, we construct a oriented cycle $D = Dir(C, C_0, C_1)$. It is straightforward to check that the oriented path in $D$ corresponding to the piece $B_1$ has $\lceil \frac{|B_1|}{2} \rceil$ forward arcs and $\lfloor \frac{|B_1|}{2} \rfloor$ backward arcs. Hence the oriented path has net length one. The oriented path corresponding to $R_2$ has $\frac{|R_2|}{2}$ forward arcs and $\frac{|R_2|}{2}$ backward arcs; the path has net length zero. Similarly, the paths in $D$ corresponding to $B_2$ and $R_1$ have net length zero and one respectively. Therefore, the cycle $D$ has net length two and by [30] or [15] $D$-COL is polynomial. Hence, $C$-COL is polynomial. ■

**Theorem 6.5.5** *Suppose $C$ is a two-edge-coloured cycle with four pieces $B_1$, $B_2$, $R_1$, and $R_2$ where*

*(i) $B_1$ is blue and of length $2b_1 + 1$,*

*(ii) $B_2$ is blue and of length $2b_2$,*

*(iii) $R_1$ is red and of length $2r_1$,*

*(iv) $R_2$ is red and of length $2r_2$.*

*Then $C$-COL is polynomial.*

**Proof** Let $G$ be an instance of $C$-COL. Let $S$ be the set of vertices in $G$ distance one in red from a mixed vertex. We reduce $C$-COL to 2SAT, where the variables correspond to the vertices of $S$ and the clauses correspond to paths between the vertices of $S$. We begin by making an observation that will simplify the description of the 2SAT instance. Suppose $x$ and $y$ are boolean variables. Then $x = y$ if and only if $(x \vee \neg y) \wedge (\neg x \vee y)$ is true; $x \neq y$ if and only if $(x \vee y) \wedge (\neg x \vee \neg y)$ is true. Therefore, we will use the clause $x = y$, (resp. $x \neq y$) to refer to the clause $(x \vee \neg y) \wedge (\neg x \vee y)$, (resp. $(x \vee y) \wedge (\neg x \vee \neg y)$).

Label the mixed vertices of $C$ with $m_1, m_2, m_3, m_4$ where $m_1$ and $m_2$ are the end-points of $B_1$, $m_1$ and $m_3$ are the end points of $R_2$. We assume without loss of generality that $r_2 \geq r_1$. There are three cases to consider.

**Case 1:** $r_2 \geq r_1 > 1$. Let $s_1, s_2, s_3, s_4$ be the vertices, in $C$, distance one in red from $m_1, m_2, m_3, m_4$ respectively. Assign the following labels to $s_1, s_2, s_3, s_4$:

| *Vertex* | *Label* |
|---|---|
| $s_1$ | 000 |
| $s_2$ | 111 |
| $s_3$ | 010 |
| $s_4$ | 110 |

This set is 2SAT-describable since the clause $(u_3 \vee \neg u_1) \wedge (\neg u_3 \vee u_2)$ is satisfied by a bit string of length three if and only if the string is one of the four labels above.

For each vertex $u \in S$ construct three variables $(u_3, u_2, u_1)$. These are the variables of the 2SAT instance. We now describe which clauses to add to the set of clauses in the 2SAT instance. For each $u \in S$ add the clause $(u_3 \vee \neg u_1) \wedge (\neg u_3 \vee u_2)$ to the set of clauses to insure $u$ is mapped to one of $\{s_1, s_2, s_3, s_4\}$. Also, if $u, v \in S$ are both adjacent to some mixed vertex $w \in V(G)$, then $u$ and $v$ must have the image in $C$. Hence add the clause $(u_1 = v_1) \wedge (u_2 = v_2) \wedge (u_3 = v_3)$. We add the other clauses based on the paths between vertices of $S$.

For $u$ and $v$ in $S$ joined by a path consisting of a single red edge, a blue path $P$, and a single red edge, add the following clause to the set of clauses:

| *Parity of P* | *Length of P* | *Clause* |
|---|---|---|
| odd | $\|P\| < 2b_1 + 1$ | $(u_1 = \neg u_1)$ |
| odd | $\|P\| \geq 2b_1 + 1$ | $(u_1 \neq v_1) \wedge (u_3 \neq v_3)$ |
| even | $\|P\| < 2b_2$ | $(u_1 = v_1) \wedge (u_2 = v_2) \wedge (u_3 = v_3)$ |
| even | $\|P\| \geq 2b_2$ | $(u_1 = v_1) \wedge (u_2 = v_2)$ |

For all $u$ and $v$ in $S$ joined by a red path $Q$, add the following clause to the set of clauses:

| *Parity of Q* | *Length of Q* | *Clause* |
|---|---|---|
| odd | $\|Q\| \geq 1$ | $(u_1 = \neg u_1)$ |
| even | $\|Q\| < 2r_1 - 2$ | $(u_1 = v_1) \wedge (u_2 = v_2) \wedge (u_3 = v_3)$ |
| even | $2r_1 - 2 \leq \|Q\| < 2r_2 - 2$ | $(u_2 = v_2) \wedge (u_3 = v_3)$ |
| even | $\|Q\| \geq 2r_2 - 2$ | $(u_3 = v_3)$ |

**Claim 6.5.5.1** *The resulting instance of 2SAT is satisfiable if and only if* $G \to C$.

**Proof of claim.** Suppose $G \to C$. Then each vertex in $S$ maps to one $\{s_1, s_2, s_3, s_4\}$. For each $u \in S$ assign $(u_3, u_2, u_1)$ the label of the vertex $s_i$ to which $u$ is mapped. This assignment is a satisfying truth assignment.

On the other hand, suppose there exists a satisfying truth assignment. We construct a function $f$ from the mixed vertices of $G$ to the mixed vertices of $C$ such that $f$ can be extended to a homomorphism of $G$ to $C$. Let $v$ be a mixed vertex in $G$. Let $u \in S$ be distance one in red from $v$. The set of variables $(u_3, u_2, u_1)$ has a truth assignment corresponding to some $s_i$. Set $f(v) = m_i$. Using the ideas from the proof of Theorem 6.1.1 one can verify that $f$ can be extended to a homomorphism $f : G \to C$. This establishes the claim.

**Case 2:** $r_2 > r_1 = 1$. Let $s_1, s_2, s_3$ be the mixed vertices distance one in red from $m_1, m_2, m_3$ respectively. Notice $s_2$ is also distance one from $m_4$. Assign the following labels to $s_1, s_2, s_3$:

| *Vertex* | *Label* |
|---|---|
| $s_1$ | 00 |
| $s_2$ | 11 |
| $s_3$ | 10 |

For each vertex $u \in S$ construct two variables $(u_2, u_1)$. These are the variables of the 2SAT instance. We now describe which clauses to add to the set of clauses in the 2SAT instance. For each $u \in S$ add the clause $(u_2 \vee \neg u_1)$. This insures that $(u_2, u_1)$

is never assigned $(0,1)$; that is, $(u_2, u_1)$ is assigned on the three labels above. Also, if $u, v \in S$ are both adjacent to some mixed vertex $w \in V(G)$, then $u$ and $v$ must have the image in $C$. Hence add the clause $(u_1 = v_1) \wedge (u_2 = v_2)$. We add the other clauses based on the paths between vertices of $S$.

For $u$ and $v$ in $S$ joined by a path consisting of a single red edge, a blue path $P$, and a single red edge, add the following clause to the set of clauses:

| *Parity of P* | *Length of P* | *Clause* |
|---|---|---|
| odd | $\|P\| < 2b_1 + 1$ | $(u_1 = \neg u_1)$ |
| odd | $\|P\| \geq 2b_1 + 1$ | $(u_1 \neq v_1) \wedge (u_2 \neq v_2)$ |
| even | $\|P\| < 2b_2$ | $(u_1 = v_1) \wedge (u_2 = v_2)$ |
| even | $\|P\| \geq 2b_2$ | $(u_2 = v_2)$ |

For all $u$ and $v$ in $S$ joined by a red path $Q$, add the following clause to the set of clauses:

| *Parity of Q* | *Length of Q* | *Clause* |
|---|---|---|
| odd | $\|Q\| \geq 1$ | $(u_1 = \neg u_1)$ |
| even | $\|Q\| < 2r_2 - 2$ | $(u_1 = v_1) \wedge (u_2 = v_2)$ |
| even | $\|Q\| \geq 2r_2 - 2$ | $(u_1 = v_1)$ |

The resulting instance of 2SAT is satisfiable if and only if $G \rightarrow C$. Suppose $f : G \rightarrow C$. Then each vertex in $S$ must map to one of $\{s_1, s_2, s_3\}$. The corresponding truth assignment is a satisfying truth assignment for the instance of 2SAT.

On the other hand, suppose there exists a satisfying truth assignment of the 2SAT instance. We construct a function $f$ from the mixed vertices of $G$ to the mixed vertices of $C$ so that $f$ can be extended to a homomorphism from $G$ to $C$. Let $v$ be a mixed vertex in $G$ distance one from $u \in S$. The set of variables $(u_2, u_1)$ has a truth assignment corresponding to some $s_i$. If it corresponds to $s_1$, then set $f(v) = m_1$. If it corresponds to $s_3$, then set $f(v) = m_3$. If it corresponds to $s_2$, then there are two cases to consider. Firstly, if $v$ is connected by a blue path to a mixed vertex, w, such that $f(w) = m_3$ then set $f(v) = m_4$. Otherwise, $v$ is connected by a blue path to mixed vertices, $w'$, such that $f(w') = m_2$ or $f(w') = m_1$ or $v$ is not joined to any mixed vertices by a blue path. In this case set $f(v) = m_2$. Using the proof of Theorem 6.1.1 one can verify that $f$ can be extended to a homomorphism $f : G \to C$.

**Case 3:** $r_2 = r_1 = 1$. Let $s_1, s_2$ be the mixed vertices distance one in red from $m_1, m_2$ respectively. Notice $s_1$ is also distance one from $m_3$ and $s_2$ is distance one from $m_4$. Assign the following labels to $s_1, s_2$:

| *Vertex* | *Label* |
|---|---|
| $s_1$ | 1 |
| $s_2$ | 0 |

For each vertex $u \in S$ construct one variable $u$. These are the variables of the 2SAT instance. We now describe which clauses to add to the set of clauses in the 2SAT instance. If $u, v \in S$ are joined to a common mixed vertex $w \in V(G)$, then $u$ and $v$ must map to the same vertex in $C$. In the two cases above we required a special clause to ensure this happened. In this final case we do not require a special clause since $u$ and $v$ a joined by a red path and that case is handled below.

The clauses are based on the paths between vertices of $S$. For $u$ and $v$ in $S$ joined by a path consisting of a single red edge, a blue path $P$, and a single red edge, add the following clause to the set of clauses:

| *Parity of P* | *Length of P* | *Clause* |
|---|---|---|
| odd | $\|P\| < 2b_1 + 1$ | $(u = \neg u)$ |
| odd | $\|P\| \geq 2b_1 + 1$ | $(u \neq v)$ |
| even | $\|P\| < 2b_2$ | $(u = v)$ |
| even | $\|P\| \geq 2b_2$ | $(u = u)$ |

For $u$ and $v$ is $S$ joined $S$ joined by a red path $Q$ add the clause $(u = v)$ to the set of clauses.

The resulting instance of 2SAT is satisfiable if and only if $G \to C$. Suppose $f : G \to C$. Then each vertex in $S$ must map to one of $\{s_1, s_2\}$. The corresponding truth assignment is a satisfying truth assignment for the instance of 2SAT.

On the other hand, suppose that the instance of 2SAT has a satisfying truth assignment. We construct a mapping $f$ from the mixed vertices of $G$ to the mixed vertices of $C$ and observe that this mapping can be extended to a homomorphism $f : G \to C$. Let $v$ be a mixed vertex in $G$ distance one in red from $u \in S$. The are four cases to consider:

- If $v$ is joined by an odd length blue path to another mixed vertex and $u$ is assigned 1, then set $f(v) = m_1$.

- If $v$ is joined by an odd length blue path to another mixed vertex and $u$ is assigned 0, then set $f(v) = m_2$.
- If $v$ is joined only by even length blue paths to another mixed vertices or is joined to no other mixed vertices and $u$ is assigned 1, then set $f(v) = m_3$.
- If $v$ is joined only by even length blue paths to another mixed vertices or is joined to no other mixed vertices and $u$ is assigned 1, then set $f(v) = m_1$.

Again, it is straightforward to check, using the proof of Theorem 6.1.1, that $f$ can be extended to a homomorphism $f : G \to C$.

This completes case 3 and the proof of the theorem. ∎

# Chapter 7

# Cliques

The results in this chapter concern the complexity of H-COL when $H$ is a clique. We begin by studying cliques with four or less vertices. Notice, that H-COL is trivial when $H$ is a clique on zero or one vertices. The classification of two-cliques is in Section 7.1. The classification of loop-free three-cliques is in Section 7.2. We also provide, in Section 7.3, a classification of three-cliques with loops allowed but we restrict attention to the case of two edge-colours. Finally, we give a classification of two-edge-coloured four-cliques with the restriction that the four-cliques are digon-free.

In Section 7.5 we study the problem for cliques larger than four-cliques. We present two infinite families of cliques for which $H$-COL is polynomial for every member of the family. The first family consists of two-edge-coloured cliques. The second family consists of digon-free, loop-free cliques. In fact, for such $k$-edge-coloured cliques, we show that for every $n \leq 2^k$ there exists an $n$-clique for which $H$-COL is polynomial and for every $n > 2^k$ any $k$-edge-coloured $n$-clique is NP-complete.

We now make an observation that reduces the number of cases that need to be considered. Let $H_1$ and $H_2$ be edge-coloured graphs with multiplicity $k$ both on the same vertex-set $V$. Further suppose there exists a permutation, $\pi$, on $\{1, 2, 3, \ldots, k\}$, such that $E_i(H_1) = E_{\pi(i)}(H_2)$. That is, $H_1$ can be obtained from $H_2$ by permuting the edge-colours of $H_2$. Then we can show that $H_1$-COL and $H_2$-COL are polynomially equivalent. Denote this permutation $H_1 = \pi(H_2)$. To see $H_1$-COL $\alpha$ $H_2$-COL, let $G$ be an instance of $H_1$-COL. Then $G \rightarrow H_1$ if and only if $\pi(G) \rightarrow H_2$. On the other hand, to see $H_2$-COL $\alpha$ $H_1$-COL, let $G$ be an instance of $H_2$-COL. Then $\pi^{-1}(G) \rightarrow H_1$ if and only if $G \rightarrow H_2$. In particular, when classifying H-COL for edge-coloured cliques of multiplicity two we can restrict our attention to cases where the number of blue edges is greater than or equal to the number of red edges.

## 7.1 Two-cliques

All the two-clique colouring problems can be reduced to 2SAT. This reduction applies to any two-clique regardless of the number of edge-colours. Label the vertices of the clique with $\{0, 1\}$. This is a 2SAT-describable set by Observation 2.3.1. For a particular edge-colour, there are eight possible edge-sets on the vertex-set $\{0, 1\}$. Namely, there are two choices (present or not) for each of the three edges, $\{00, 01, 11\}$. For each of these possible edge-sets the obvious partition, $\{0, 1\} = \{0\} \cup \{1\}$, satisfies the conditions in Theorem 2.3.3 and each edge-set is 2SAT amiable. Hence we have the following theorem.

**Theorem 7.1.1** *Let $H$ be an edge-coloured clique on two vertices. Then $H$-COL is solvable in polynomial time.*

## 7.2 Loop-Free Three-Cliques.

In the study of graphs and of digraphs, the classification of the complexity of $H$-COL is completely determined when $H$ contains a spanning clique. See [19] and [1]. These classifications are given in terms of the existence of certain subgraphs. In the case of graphs, if $H$ contains a subgraph which is a $K_3$, then the problem is NP-complete and if it does not contain such a subgraph it is polynomial. In the case of digraphs, if $H$ is a semicomplete digraph (at least one arc between any pair of vertices) and $H$ contains at least two directed cycles, then $H$-COL is NP-complete and if $H$ contains zero or one directed cycles, then $H$-COL is polynomial. It would be nice to have such a subgraph characterization for edge-coloured cliques as well, but it seems unlikely even for the three-clique problem. Consider the sequence of edge-coloured graphs in Figure 7.1. Each edge-coloured graph is a subgraph of the following graph. However, the complexity alternates between polynomial and NP-complete, demonstrating that a subgraph characterization is impossible.

In this section we restrict our attention to loop-free three-cliques. For this restricted class, we show that the complexity of $H$-COL is completely determined by the existence or non-existence of certain subgraphs.

We now use 2SAT to show a particular class of three-cliques is polynomial. Let $H$ be an edge-coloured three-clique where the vertices of $H$ have been labelled with the bit-strings $\{00, 01, 11\}$. This is a 2SAT-describable set by Lemma 2.3.2. Initially we will restrict our attention to those three-cliques $H$ that do not contain a monochromatic $K_3$. (The existence of a monochromatic $K_3$ immediately implies the problem

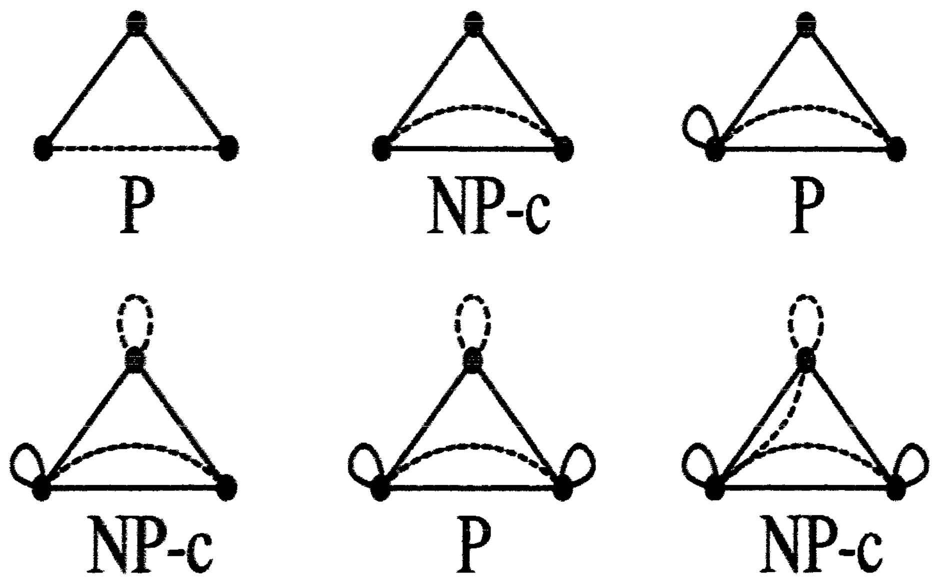


Figure 7.1: An alternating sequence of $K_3$'s

is NP-complete.) Given that the vertices are labelled, there are six possible edge-configurations that a particular edge-colour can take. (Recall we are restricting our attention to loop-free and monochromatic $K_3$-free configurations). The possibilities are listed below and drawn in Figure 7.2.

| *Name* | *Edge set* |
|---|---|
| $C_1$ | $\{\{11, 00\}\}$ |
| $C_2$ | $\{\{11, 01\}\}$ |
| $C_3$ | $\{\{01, 00\}\}$ |
| $C_4$ | $\{\{01, 00\}, \{00, 11\}\}$ |
| $C_5$ | $\{\{01, 11\}, \{00, 11\}\}$ |
| $C_6$ | $\{\{11, 01\}, \{01, 00\}\}$ |

The following lemma tells us that the only possibly 'bad' configuration for the edges is $C_6$.

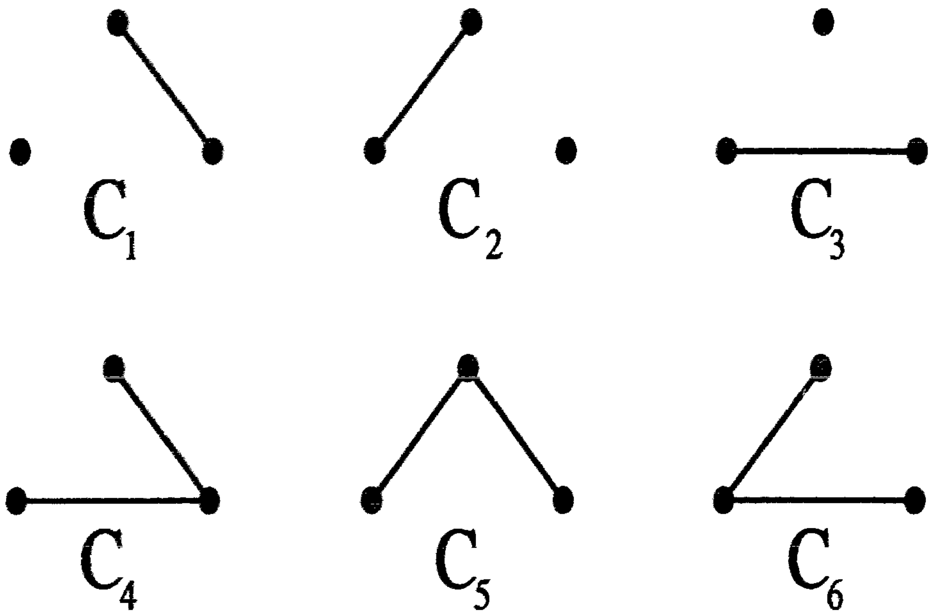


Figure 7.2: Possible configuration for each edge-colour-class.

**Lemma 7.2.1** *Suppose $H$ is a loop-free, edge-coloured three-clique containing no monochromatic $K_3$. If the vertices of $H$ can be labelled with $\{00, 01, 11\}$ such that each edge-colour-class is one of the configurations $C_1, \ldots, C_5$ above, then $H$-COL is polynomial.*

**Proof** We use a reduction to 2SAT to solve the problem. We have already observed that the set of bit-strings used to label $H$ is 2SAT-describable. Therefore, we only need to give clauses for each configuration $C_1, \ldots, C_5$. These clauses are given below.

| *Configuration* | *Clause* |
|---|---|
| $C_1$ | $(v_2 \vee u_2) \wedge (\neg v_2 \vee \neg u_2) \wedge (v_1 \vee u_1) \wedge (\neg v_1 \vee \neg u_1)$ |
| $C_2$ | $(v_2 \vee u_2) \wedge (\neg v_2 \vee \neg u_2) \wedge (v_1) \wedge (u_1)$ |
| $C_3$ | $(\neg u_2) \wedge (\neg v_2) \wedge (v_1 \vee u_1) \wedge (\neg v_1 \vee \neg u_1)$ |
| $C_4$ | $(v_1 \vee u_1) \wedge (\neg v_1 \vee \neg u_1)$ |
| $C_5$ | $(v_2 \vee u_2) \wedge (\neg v_2 \vee \neg u_2)$ |

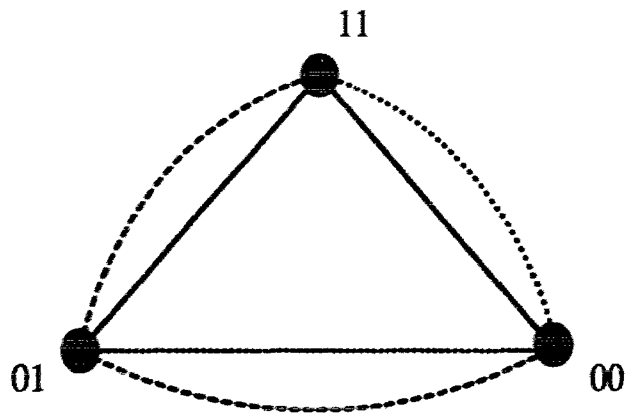


Figure 7.3: The edge-coloured graph $H_6$.

The verification that these clauses are correct is straightforward. The result follows. ∎

It may seem possible that another labelling of $H$ with different bit-strings could allow us to use 2SAT for $C_6$. However, the following lemma shows that configuration $C_6$ is indeed difficult. Let $H_6$ be the edge-coloured graph with vertex-set $\{00, 01, 11\}$ and three edge-colours: blue, red, and green. Let the blue edge-set be $\{\{01,11\},\{11,00\}\}$, the red edge-set be $\{\{01,11\},\{01,00\}\}$, and the green edge-set be $\{\{01,00\},\{11,00\}\}$. A picture of $H_6$ is in Figure 7.3. Blue edges are solid, red edges are dashed, and green edges are dotted.

**Lemma 7.2.2** *Let $H$ be a loop-free three-clique. Suppose there are three edge-colours red, green, and blue such that the subgraph induced by these three colours is precisely $H_6$. Then $H$-COL is NP-complete.*

**Proof.** We use two applications of the indicator construction to show $H_6$-colouring is NP-complete. Let $I_1$ be a path of length five, with edges $\{e_1, e_2, e_3, e_4, e_5\}$. The ordering on the edges is the natural ordering. All edges are blue with the exception

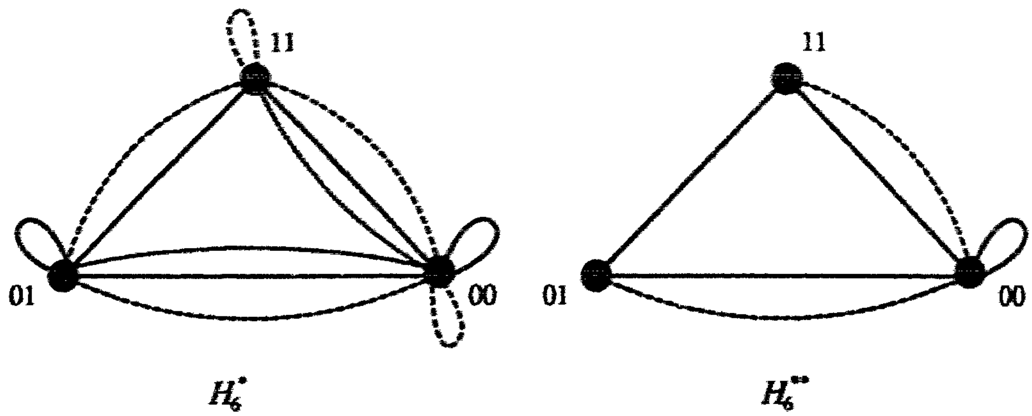


Figure 7.4: The edge-coloured graphs $H_6^*$ and $H_6^{**}$.

of $e_3$ which is red. Vertices $i_1$ and $j_1$ are the endpoints of the path. Similarly, let $I_2$ be a path of length five with the outer four edges red and the center edge green. The end points of $I_2$ are $i_2$ and $j_2$. Finally, let $I_3$ be a single green edge with end points $i_3$ and $j_3$. We consider blue to be colour one, red to be colour two, and green to be colour three. The result of the indicator construction with respect to $(I_1, i_1, j_1), (I_2, i_2, j_2), (I_3, i_3, j_3)$ is the edge-coloured graph $H_6^*$. The vertices of $H_6^*$ are $\{00, 01, 11\}$. The blue edge-set is $\{\{00, 00\}, \{01, 00\}, \{01, 11\}, \{11, 00\}, \{01, 01\}\}$. The red edge-set is $\{\{01, 00\}, \{01, 11\}, \{11, 00\}, \{00, 00\}, \{11, 11\}\}$. The green edge-set is $\{\{01, 00\}, \{00, 11\}\}$. A picture of $H_6^*$ is in figure 7.4. Blue edges are solid, red edges are dashed, and green edges are dotted.

We use the indicator construction on $H_6^*$. Let $I_1$ be the digon on vertices $\{i_1, j_1\}$ with a blue and a red edge, $i_1 j_1$. Let $I_2$ be the single green edge on vertices $i_2, j_2$. The result of the indicator construction with respect to $(I_1, i_1, j_1), (I_2, i_2, j_2)$ is the graph $H_6^{**}$. The blue edges are $\{\{01, 11\}, \{11, 00\}, \{01, 00\}, \{00, 00\}\}$ and the red edges are $\{\{01, 00\}, \{11, 00\}\}$. A picture of $H_6^{**}$ is in Figure 7.4. The $H_6^{**}$-colouring problem is NP-complete. The proof of this appears in the next section. This implies both $H^*$-COL and $H$-COL are NP-complete. ∎

The previous two lemmas now allow use to classify all loop-free three-clique colouring problems.

**Theorem 7.2.3** *Let $H$ be a loop-free three-clique. If $H$ contains a monochromatic triangle or three edge-colours that induce a copy of $H_6$, then $H$-COL is NP-complete; otherwise, $H$-COL is polynomial.*

**Proof.** If $H$ contains a monochromatic triangle or $H_6$, then restrict the input to the edge-colours of this subgraph. In these cases $H$-COL is NP-complete.

Otherwise, label the vertices of $H$ with the bit-strings $\{00, 01, 11\}$ such that each edge-colour induces one of the configurations $C_1$ to $C_5$. This is always possible provided $H$ does not contain three colours that induce $H_6$. Use Lemma 7.2.1 to conclude the problem is polynomial. ■

## 7.3 Two-Edge-Coloured Three-Cliques

In this section we allow loops but restrict the multiplicity of $H$ to two. Each $H$-colouring problem (for $H$ a two-edge-coloured three-clique) is classified as either NP-complete or we present a polynomial time algorithm. We label the vertices of $H$ with $\{0, 1, 2\}$. If we ignore loops, then there are essentially six two-edge-coloured cliques. Since loops will be introduced later we can no longer exclude monochromatic triangles. As mentioned above we need only consider the case where the number of blue edges is greater than or equal to the number of red edges. The list below and Figure 7.5 describe these six cliques.

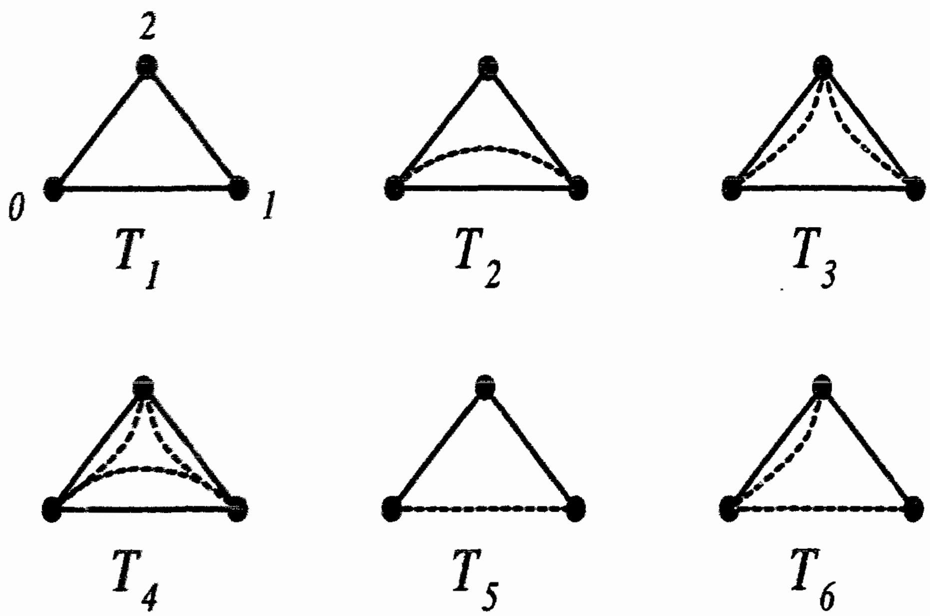


Figure 7.5: All six loop-free, two-edge-coloured cliques on three vertices.

| *Name* | *Blue Edges* | *Red Edges* |
|---|---|---|
| $T_1$ | $\{01, 12, 20\}$ | $\{\}$ |
| $T_2$ | $\{01, 12, 20\}$ | $\{01\}$ |
| $T_3$ | $\{01, 12, 20\}$ | $\{20, 21\}$ |
| $T_4$ | $\{01, 12, 20\}$ | $\{01, 12, 20\}$ |
| $T_5$ | $\{20, 21\}$ | $\{01\}$ |
| $T_6$ | $\{20, 21\}$ | $\{20, 01\}$ |

We now consider the complexity of H-COL when $H$ is one of the six edge-coloured graphs above. We also consider all possible additions of loops and classify those problems as well. To avoid trivialities, we will never have a red and a blue loop on the same vertex because any two-edge-coloured graph will map to such a double loop. That is, $G$ is a YES instance if and only if $G$ has only red and blue edges. Also, in the following we will use the term $T_i^+$ to denote any two-edge-coloured clique on three vertices obtained from $T_i$ by adding a (possibly empty) set of loops to the edge-set.

The indicator construction will be used several times. We first describe our set of indicators. We use either single edges or paths of length three. In all cases, the specified vertices are the end-points of the edge and path respectively. Recall that the blue edges are $E_1$ and the red edges are $E_2$. Hence the images of indicator $I_1$ correspond to the blue edges in $H^*$ and the images of the indicator $I_2$ correspond to the red edges in $H^*$.

| *Name* | *Description* |
|---|---|
| $R$ | Single red edge. |
| $B$ | Single blue edge. |
| $D$ | Two vertices joined by a red and a blue edge. |
| $RRR$ | Path of length three of all red edges. |
| $BBB$ | Path of length three of all blue edges. |
| $RBR$ | Path of a red edge, blue edge, and red edge. |
| $BRB$ | Path of a blue edge, red edge, and blue edge. |
| $RDR$ | Path of a red edge, $D$ from above, and red edge. |
| $BDB$ | Path of a blue edge, $D$ from above, and blue edge. |

### 7.3.1 The Complexity of $T_1^+$-Colouring

We begin by showing $T_1^+$-colouring is NP-complete if $T_1^+$ contains no blue loops; that is, the graph induced by the blue edges is a $K_3$. We reduce $K_3$-COL to $T_1^+$-COL. Let $G$ be an instance of $K_3$-COL. Let $G_b$ be the two-edge-coloured graph on $V(G)$ with $E_1(G_b) = E(G)$ and $E_2(G_b) = \phi$. We claim $G \to K_3$ if and only if $G_b \to T_1^+$.

Informally we say we reduce $T_1^+$-COL to $K_3$-COL by restricting the input to edge-coloured graphs with only blue edges.

On the other hand, if $T_1^+$ has a blue loop we can use the Dominating Loop Lemma (Lemma 2.4.3) to reduce the problem to a two vertex clique problem. These are all polynomial by Theorem 7.1.1.

**Observation 7.3.1** *The problem $T_1^+$-COL is NP-complete if $T_1^+$ has no blue loops and is polynomial otherwise.*

### 7.3.2 The Complexity of $T_4^+$-Colouring

While many will argue that the $T_2^+$ classification should come after the $T_1^+$ classification we require these results in the following subsections. Hence, the possibly unaesthetic ordering must be endured.

Consider any $T_4^+$. Since we are assuming that no vertex of the $T_4^+$ has both a red loop and a blue loop, the result of the indicator construction with respect to $(D, i, j)$ is a $T_1$. Since $T_1$-COL is NP-complete, all $T_4^+$-colouring problems are NP-complete.

**Observation 7.3.2** *Each problem $T_4^+$-COL is NP-complete.*

### 7.3.3 The Complexity of $T_2^+$-Colouring

Consider the following cases:

**Case $T_2^+$-1: No blue loop.** We can restrict the input to blue edges only. This problem is NP-complete since the blue edges induce $T_1$ and $T_1$-COL is NP-complete.

**Case $T_2^+$-2: A blue loop on vertex 2.** The problem reduces via the Dominating Loop Lemma (Lemma 2.4.3) to a two vertex clique problem and hence is polynomial.

We can now assume that $T_2^+$ has at least one blue loop on vertices $\{0,1\}$ and either a red loop or no loop on vertex 2.

**Case $T_2^+$-3: No loop on vertex 2. Blue loop on vertex 0 and/or 1.** These graphs retract to a two vertex clique and therefore are polynomial.

**Case $T_2^+$-4: Red loop on vertex 2. Blue loops on vertices 0 and 1.** This graph is polynomial but we have not found a generic tool to show this. Hence, we present a special algorithm for this edge-coloured graph. Call this edge-coloured graph $T_2^+$-4. Let $G$ be an instance of $T_2^+$-4-COL. Consider the red components of $G$. The nonbipartite red components must all map to the red loop at vertex 2. Hence there can be no blue edges in a nonbipartite red component nor can there be any blue edges joining two vertices each in two separate nonbipartite red components. We claim this condition is necessary and sufficient for $G$ to map to $T_2^+$-4. The necessity of the condition has just been demonstrated. Suppose $G$ satisfies the condition. Then we map $G$ to $T_2^+$-4 as follows:

1. Map all vertices belonging to a red nonbipartite component to vertex 2.

2. Map each red bipartite component to $\{0,1\}$.

3. Map all remaining vertices to vertex 0.

The proof that this is a homomorphism is straightforward. Observe that there are blue edges in $T_2^+$-4 between all pairs of vertices from $T_2^+$-4 with the single exception that vertex 2 does not have a blue edge to itself (i.e. a blue loop). However, the condition on $G$ says the preimage of 2 (i.e. the vertices mapped in step 1) are blue edge free.

**Case $T_2^+$-5: Red loop on vertex 2. Blue loop on exactly one of vertices 0 and 1.** Without loss of generality assume there is a blue loop on vertex 0 and no blue loop on vertex 1. Let $I_1$ be the $B$ indicator and $I_2$ be the $RBR$ indicator. The indicator construction with respect to $((I_1, i_1, j_1), (I_2, i_2, j_2))$ results in a $T_4^+$. Hence, $T_2^+$-5 colouring is NP-complete.

### 7.3.4 The Complexity of $T_3^+$-Colouring

**Case $T_3^+$-1: No loops on vertices 0 or 1.** If there is no blue loop on vertex 2, then we can restrict the input to blue edges and conclude the problem is NP-complete.

If on the other hand, there is a blue loop on vertex 2, then we use Lemma 2.4.1. Let $H_1$ be the edge-coloured graph induced by vertices 0 and 1. Let $H_2$ be the blue loop on vertex 2. Notice $H_1 \to H_2$ as required and $T_3^+$ is the join of $H_1$ and $H_2$ with respect to red and blue. Since $H_1$ join $H_1$ is a monochromatic $K_4$ and $K_4$-COL is NP-complete, we conclude that $T_3^+$-1-COL is NP-complete.

**Case $T_3^+$-2: Blue loop on at least one of $\{0,1\}$. No red loop on either $\{0,1\}$.** Consider the congruence with two classes $S_0 = \{0,1\}$ and $S_1 = \{2\}$. This congruence induces a retraction to a two-clique. Hence, the problem is polynomial.

**Case $T_3^+$-3: Red loop on at least one of $\{0,1\}$.** Assume without loss of generality that there is a red loop on vertex 0. Vertex 1 may have a red loop, a blue loop, or no loop. Use the indicator $B$ for and $I_1$ and $RDR$ for $I_2$. The indicator construction with respect to $((I_1, i_1, j_1), (I_2, i_2, j_2))$ produces a $T_4^+$ and therefore the problem is NP-complete.

### 7.3.5 The Complexity of $T_5^+$-Colouring

**Case $T_5^+$-1: No blue loop.** Let $H_1$ be the edge-coloured graph induced by $\{0,1\}$ and let $H_2$ be the edge-coloured graph induced by $\{2\}$. Observe that both $H_1$-COL and $H_2$-COL are polynomial. We now observe that $T_5^+$-1 satisfies the conditions of Lemma 3.5.1. Hence, all $T_5^+$-1 problems are polynomial.

**Case $T_5^+$-2: Blue loop on vertex 2.** Use the Dominating Loop Lemma (Lemma 2.4.3) to reduce the problem to a two-clique-colouring problem. Hence, all $T_5^+$-2-COL problems are polynomial.

**Case $T_5^+$-3: No loop on vertex 2. Exactly one blue loop on vertices $\{0,1\}$. Either 0 or 1 red loops on vertices $\{0,1\}$.** Without loss of generality assume there is a blue loop on vertex 0. There are two possibilities for vertex 1. Either vertex 1 has no loop or vertex 1 has a red loop. Call the former case "subcase A" and the latter case "subcase B". Both these problems reduce to 2SAT. Label the vertices with bit strings of length two as follows:

| *Vertex* | *Label* |
|---|---|
| 0 | 00 |
| 1 | 10 |
| 2 | 01 |

This is 2SAT-describable by Lemma 2.3.2. Unfortunately, the edge-coloured graphs $T_5^+$-3 are not 2SAT amiable. However, we can come up with a set of *ad hoc* clauses. Given an instance, $G$, of $T_5^+$-3-COL, we construct an instance, $S$, of 2SAT as follows. For each vertex $u$ in $G$, there correspond two variables $u_1$ and $u_2$ in $S$. The clauses of $S$ are defined below.

The clauses in subcase A are:

| *Edge* | *Clause* |
|---|---|
| Blue edge $uv$ | $(u_1 \vee \neg v_2) \wedge (\neg u_2 \vee v_1) \wedge (\neg u_1 \vee \neg v_1)$ |
| Red edge $uv$ | $(u_2 \vee v_2) \wedge (\neg u_2 \vee \neg v_2) \wedge (\neg v_1) \wedge (\neg u_1)$ |

The clauses in subcase B are:

| *Edge* | *Clause* |
|---|---|
| Blue edge $uv$ | $(u_1 \vee \neg v_2) \wedge (\neg u_2 \vee v_1) \wedge (\neg u_1 \vee \neg v_1)$ |
| Red edge $uv$ | $(u_2 \vee v_2) \wedge (\neg v_1) \wedge (\neg u_1)$ |

Suppose a satisfying truth assignment exists for $S$. Because the vertices have been labelled with a 2SAT-describable set, we can assume that each pair of variables, $(u_2, u_1)$, has been assigned $(0,0), (0,1)$, or $(1,0)$. Therefore, given an edge $uv$, the variables $(u_2, u_1, v_2, v_1)$ can take on nine possible values. Notice the blue edge-clause in both subcases in the same. We call this the Blue Clause. The red edge clause in subcase A (resp. subcase B) is called Red Clause A (resp. Red Clause B).

| *Value of* $(u_2, u_1, v_2, v_1)$ | *Value of Blue Clause* | *Red Clause A* | *Red Clause B* |
|---|---|---|---|
| (0,0,0,0) | True | False | False |
| (0,0,0,1) | True | False | False |
| (0,0,1,0) | False | True | True |
| (0,1,0,0) | True | False | False |
| (0,1,0,1) | False | False | False |
| (0,1,1,0) | True | False | False |
| (1,0,0,0) | False | True | True |
| (1,0,0,1) | True | False | False |
| (1,0,1,0) | False | False | True |

It is now easy to see, there is a satisfying truth assignment for $S$ if and only if there is a homomorphism of $G$ to $T_5^+$-3.

**Case $T_5^+$-4: No loop on 2. Blue loop on both vertex 0 and on vertex 1.** Let $I_1$ be $BRB$ and $I_2$ be $B$. The indicator construction with respect to $((I_1, i_1, j_1), (I_2, i_2, j_2))$ produces an edge-coloured graph $T_3^+$-3. Therefore, all problems $T_5^+$-4-COL are NP-complete.

**Case $T_5^+$-5: Blue loop on at least one of vertices $\{0, 1\}$. Red loop on vertex 2.** Suppose without loss of generality there is a blue loop on vertex 0. Initially assume there is no loop on vertex one. Use the indicator $BR$ consisting of the edge-coloured graph on the vertex-set $\{i, x, j\}$ with $ix$ a blue edge and $xj$ a red edge. The indicator construction described in Chapter One requires that all indicators have an automorphism that maps $i$ to $j$ and $j$ to $i$. This condition ensures that the result of the indicator construction has undirected edges. If the automorphism condition is removed, then the result of the indicator construction in general will have directed edges. See [25] for more details.

The indicator construction with respect to $(BR, i, j)$ results in a semicomplete digraph with two directed cycles. The $H$-colouring problem for such a graph $H$ is proved to be NP-complete in [2]. If we add a loop of any colour to vertex 1, the result is an increase of arcs in the result of the indicator construction. Nonetheless, the result is still a semi-complete digraph with at least two directed cycles. Hence, all $T_5^+$-5-COL problems are NP-complete.

### 7.3.6 The Complexity of $T_6^+$-Colouring

**Case $T_6^+$-1: There are no loops.** Label the vertices with bit-strings of length two as follows:

| *Vertex* | *Label* |
|---|---|
| 0 | 00 |
| 1 | 10 |
| 2 | 11 |

This set is 2SAT-describable by Lemma 2.3.2. Also the labeling satisfies the requirements of Theorem 2.3.3, proving the edge-coloured graph is 2SAT amiable. Therefore, this problem is polynomial.

**Case $T_6^+$-2: Blue loop on vertex 2. No red loop on vertex 1.** In this case, there is a retraction to the subgraph induced by vertices $\{0,2\}$ obtained by mapping vertex 1 to vertex 2. This reduces the problem to a two-clique-colouring problem which is polynomial.

**Case $T_6^+$-3: Blue loop on vertex 2. Red loop on vertex 1.** If there is also a red loop on vertex 0, then the edge-coloured graph retracts to the subgraph induced by $\{0,2\}$. The problem is polynomial. If there is not a red loop on vertex 0, then we label the vertices with the following bit-strings of length 2:

| *Vertex* | *Label* |
|---|---|
| 0 | 01 |
| 1 | 00 |
| 2 | 10 |

We again observe this a 2SAT-describable set. Let $G$ be an instance of H-COL. We construct an instance of 2SAT from $G$ as follows:

| *Edge* | *Clause* |
|---|---|
| Blue edge $uv$ | $(u_2 \vee v_2)$ |
| Red edge $uv$ | $(\neg u_2 \vee \neg v_2) \wedge (\neg u_2 \vee v_1) \wedge (u_1 \vee \neg v_2) \wedge (\neg u_1 \vee \neg v_1)$ |

As above, one can verify this instance of 2SAT is a YES instance if and only if $G \to H$.

**Case $T_6^+$-4: Red loop on vertex 2 and vertex 0. No blue loop on vertex 1.** This edge-coloured graph retracts to the subgraph induced by $\{0, 2\}$. The problem is polynomial.

**Case $T_6^+$-5: Red loop on vertex 2. Not case $T_6^+$-4.** If there is a blue loop on either vertex 0 or vertex 1 or both, then let $I_1$ be $RDR$ and $I_2$ be $B$. Observe that neither of these indicators produce a loop in $H^*$ that is not present in $H$. Hence, $H^*$ does not contain a double loop (red and blue) on any vertex since $H$ does not contain a double loop. The result of the indicator construction with respect to $(I_1, i_1, j_1), (I_2, i_2, j_2)$ is an edge-coloured graph from case $T_3^+$-2. Thus, the problem is NP-complete.

Now assume there is not a blue loop on either vertex 0 nor vertex 1. That is, there are no loops on 0 or 1 or a red loop on vertex 1 and no loop on vertex 0. Let $I_1$ be $RDR$ and let $I_2$ be $B$. The result of the indicator construction with respect to $(I_1, i_1, j_1)$, $(I_2, i_2, j_2)$ is an edge-coloured graph from case $T_3^+$-1. This is NP-complete.

**Case $T_6^+$-6: No loop on vertex 2. Red loop on vertex 0.** If there is not a blue loop on vertex 1, then the edge-coloured graph retracts to the subgraph induced by $\{0,2\}$ and the problem is polynomial. If there is a blue loop on vertex 1, then by switching red edges for blue and vice versa we are in case $T_6^+$-3.

**Case $T_6^+$-7: No loop on vertex 2. No red loop on vertex 0.** If there is a red loop on vertex 1 and no blue loop on 0, then we use the same reduction to 2SAT as in case $T_6^+$-3, except we use the following clause for the blue edges:

$$(u_2 \vee v_2) \wedge (\neg u_2 \vee \neg v_2)$$

Now we can assume there is no red loop on 0 or 1, but there is a blue loop on 0 and/or 1. In this case, switch blue edges for red edges and vice versa. This result is a $T_6^+$ with a red loop on 1 and/or 2. A red loop on vertex 2 is either case $T_6^+$-4 or case $T_6^+$-5. If there is not a red loop on 2, but there is a red loop on vertex 1, then we are in case $T_6^+$-6 or the first part of the present case.

This completes the classification of the $T_6^+$ problems and thus also completes our classification of two-edge-coloured three-cliques (with loops allowed).

## 7.4 Two-Edge-Coloured Four-Cliques

The amount of work required to classify all two-edge-coloured three-cliques suggests that classifying all the four-cliques might require many more hours and hundreds more pages. Instead of making a career out of the four-clique problem, we will concentrate on the special case when the four-clique does not contain a digon, i.e. a pair of vertices $u$ and $v$ joined in both blue and red.

Given a four-clique, let the vertices be $\{0,1,2,3\}$. We consider all two-edge-colour four-cliques with the number of blue edges greater than or equal than the number

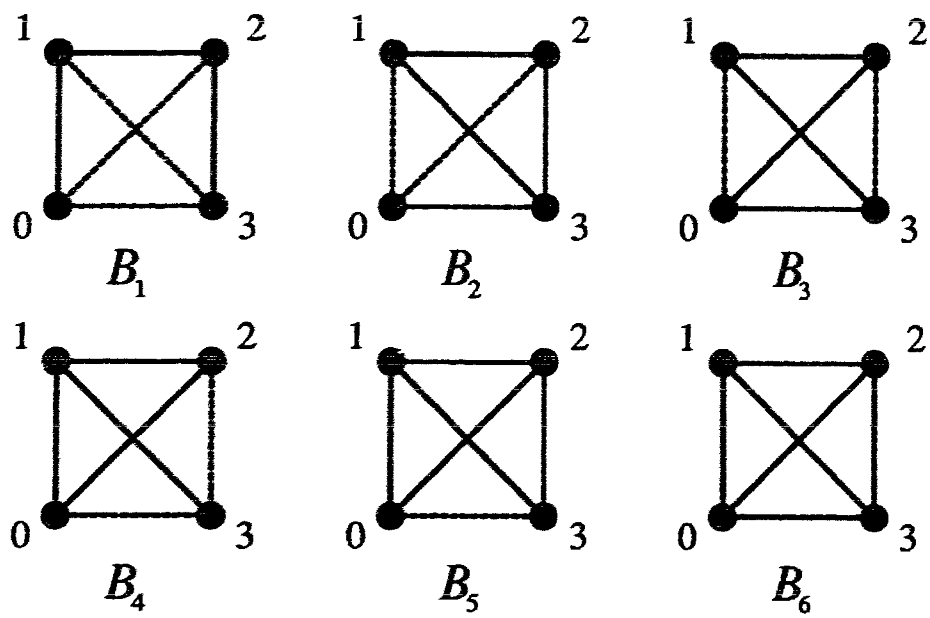


Figure 7.6: All digon-free two-edge-coloured four-cliques.

of red edges. As before, this covers all possible cases by symmetry. The following table and figure contain descriptions of all digon-free two-edge-coloured four-cliques without loops. The red edge-set is listed. The blue edge-set is the complement of the red edge-set. As in the case for three-cliques, the symbol $B_1^+$ refers to any edge-coloured four-clique obtained by adding loops to the edge-coloured graph $B_1$.

| *Name* | *Red Edge-Set* |
|---|---|
| $B_1$ | $\{13, 30, 02\}$ |
| $B_2$ | $\{01, 02, 03\}$ |
| $B_3$ | $\{10, 23\}$ |
| $B_4$ | $\{03, 23\}$ |
| $B_5$ | $\{03\}$ |
| $B_6$ | $\{\}$ |

### 7.4.1 The classification of $B_1^+$-colouring

**Theorem 7.4.1** *All $B_1^+$-colouring problems are NP-complete.*

**Proof.** Let $RB$ be the edge-coloured graph on three vertices, $\{i, m, j\}$, where $im$ is red and $mj$ is blue. The result of the indicator construction on $B_1$ with respect to $(RB, i, j)$ is a directed graph (multiplicity one). This directed graph $B_1^*$ is loop-free since $B_1$ is digon-free. Moreover, $B_1^*$ is a semicomplete digraph with two directed cycles. A result in [2] states that $H$-COL is NP-complete when $H$ is a semicomplete digraph containing two or more cycles. Therefore, $B_1$-COL is NP-complete.

Let $Z$ be any edge-coloured four-clique obtained by adding loops to $B_1$. The edge-coloured graph $Z^*$ will contain $B_1^*$ as a subgraph. That is, $Z$ is a semicomplete digraph with at least two directed cycles. We must ensure that $Z^*$ does not contain any loops. The existence of a loop in $Z^*$ implies the existence of a digon or a vertex with both a blue and red loop on it in $Z$. We have assumed that neither of these situations occurs. Therefore, all $B_1^+$-colouring problems are NP-complete. ∎

### 7.4.2 The classification of $B_2^+$-colouring

**Case $B_2^+$-1: No blue loops on vertices 1, 2 or 3.** If vertex 0 does not have a red loop, then let J be the edge-coloured graph consisting of a single red edge with end points $k_1$ and $j$. Label vertex 0 in the $B_2^+$ with $h_1$. The result of the subindicator construction with respect to $(J, j, k_1)$ is the subgraph induced by $\{1, 2, 3\}$. This is a $T_1^+$ without blue loops and is NP-complete.

If vertex 0 does have a red loop, then use the Dominating Loop Lemma (Lemma 2.4.3) to remove vertex 0. Again, vertices $\{1, 2, 3, \}$ induce a $T_1^+$ without blue loops.

The problem is NP-complete.

**Case $B_2^+$-2: Vertices $\{1, 2, 3\}$ contain at least one blue loop and at most one red loop.** Without loss of generality assume there is a blue loop on vertex 1 and no red loop on vertex 2. Then the $B_2^+$ retracts to the subgraph induced by $\{0, 1, 3\}$ by mapping vertex 2 to vertex 1. This problem is in case $T_5^+$-3 and is polynomial.

**Case $B_2^+$-3: Vertices $\{1, 2, 3\}$ contain exactly one blue loop and two red loops.** Without loss of generality assume there is a blue loop on vertex 1 and a red loop on each of vertices 2 and 3. If there is a red loop on vertex 0 as well, then use the Dominating Loop Lemma (Lemma 2.4.3) to reduce to problem to the subgraph induced by $\{1, 2, 3\}$. This problem is a $T_5^+$-2 problem that is polynomial.

If there is a blue loop on vertex 0, then again let $RB$ be the edge-coloured graph on $i, k, j$ with $ik$ red and $kj$ blue. The result of the indicator construction with respect to $(RB, i, j)$ is a semicomplete digraph with at least two directed cycles. Therefore, the problem is NP-complete.

If there is no loop on vertex 0, then let $I_1$ be the single blue edge $B$ with end points $i_1$ and $j_1$, and let $I_2$ be $RRR$, the red path of length three with end points $i_2$ and $j_2$. Let $Z^*$ be the result of the indicator with respect to $(B, i_1, j_2), (RRR, i_2, j_2)$. The blue edge-set of $Z^*$ is the same as the $B_2^+$-3; $\{11, 12, 13, 23\}$. The red edge-set is all possible edges (and loops) with the exception of 11. We claim $Z^*$-COL is NP-complete and hence the original problem is NP-complete. To see $Z^*$-COL is NP-complete use the single indicator $D$ consisting of a red-blue digon. The result is a loop free $T_1$ which is NP-complete.

### 7.4.3 The classification of $B_3^+$-colouring

**Case $B_3^+$-1: No blue loops.** Using Lemma 3.5.1 the problem reduces into two polynomial problems. Hence $B_2^+$-1-COL is polynomial.

**Case $B_3^+$-2: At least one blue loop.** Let $Z$ be the edge-coloured graph obtained by adding exactly one blue loop to $B_3$. Observe, $Z$ is unique up to isomorphism. Again let $RB$ be the edge-coloured path on three vertices consisting of a red edge followed by a blue edge. The result of the indicator construction with respect to $(RB, i, j)$ is a a digraph whose core is a semicomplete digraph with at least two directed cycles. As argued above, adding loops to $Z$ will not change the complexity. Therefore, all edge-coloured graphs in case $B_2^+$-2 are NP-complete.

### 7.4.4 The classification of $B_4^+$-colouring

**Case $B_4^+$-1: No blue loop on 0, 1, or 2.** Let $J$ be the edge-coloured graph on vertices $\{k_1, i, j\}$ where $k_1 i$ is red and $ij$ is blue. Label vertex 3 in $B_3$ with $h_1$. The result of applying the subindicator construction with respect to $(J, j, k_1)$ is the subgraph of $B_4$ induced by $\{0, 1, 2\}$. Vertex 3 can not belong to this edge-coloured graph since $B_4$ does not contain any digons. By assumption, there are no blue loops on 0, 1, or 2. Therefore, the colouring problem for $B_4^-$ is NP-complete. Observe adding loops to vertex 3 or adding red loops to vertices $\{0, 1, 2\}$ does not change the complexity. This implies all $B_4^+$-1-colouring problems are NP-complete.

**Case $B_4^+$-2: Blue loop on 0, 1, or 2.** If there is a blue loop on vertex 1, the problem reduces via the Dominating Loop Lemma (Lemma 2.4.3) to a $T_5^+$-colouring problem. Therefore, assume there is a blue loop on 0 or 2. By symmetry, we can assume that vertex 0 has a blue loop on it. If there is no red loop on vertex 2, then the edge-coloured graph retracts to the subgraph induced by $\{0, 1, 3\}$ by identifying vertices 0 and 2. This is a $T_5^+$-colouring problem.

Therefore, assume there is a blue loop on vertex 0 and a red loop on vertex 2. If there is a blue loop on vertex 3, we use a subindicator to isolate vertices$\{0, 2, 3\}$. Let $J$ be the coloured path on $\{k_1, i, j\}$ with $k_1 i$ blue and $ij$ red. Label vertex 1 in $B_4^+$-2 with $h_1$. The result of the subindicator construction with respect to $(J, j, k_1)$ is the subgraph of $B_4^+$-2 induced by $\{0, 2, 3\}$. This is a $T_5^+$ colouring problem that is NP-complete.

Hence we may assume there is not a blue loop on vertex 1 nor on vertex 3. We are still assuming there is a blue loop on 0 and a red loop on 2. Suppose either 1 or 3 has a red loop. Assume vertex 1 has a red loop, then using the indicator $RB$ from above, we construct a semicomplete digraph with two directed cycles. Again by [2] this implies the problem is NP-complete.

Lastly assume vertex 1 has no loop. This leaves two cases. Either vertex 3 has a red loop or vertex 3 has no loop. Both these problems are polynomial via a 2SAT reduction. We begin by labeling the vertices of $B_3$-2 with the following bit-strings.

| *Vertex* | *Label* |
|---|---|
| 0 | 111 |
| 1 | 101 |
| 2 | 011 |
| 3 | 010 |

This set is a 2SAT-describable set. The clause $(u_3 \vee u_2) \wedge (\neg u_3 \vee u_1)$ has three variables and is satisfied if and only if the three variables take on the values in one of the four bit-strings above.

We now give the clauses needed to describe the edges. The verification that these clauses are correct is straight forward. In the case that there is no loop on vertex 3, we use the following clauses.

| *Edge Set* | *Clause* |
|---|---|
| Blue edges | $(u_3 \vee v_3) \wedge (u_2 \vee v_2) \wedge (u_1 \vee \neg v_2) \wedge (\neg u_2 \vee v_1)$ |
| Red edges | $(u_2) \wedge (v_2) \wedge (\neg u_3 \vee \neg v_3) \wedge (\neg u_1 \vee \neg v_3) \wedge (\neg v_1 \vee \neg u_3)$ |

In the case there is a red loop on vertex 3, we use the following clauses. Note the two sets are identical except for the final disjunction on the red edges below.

| | |
|---|---|
| Blue edges | $(u_3 \vee v_3) \wedge (u_2 \vee v_2) \wedge (u_1 \vee \neg v_2) \wedge (\neg u_2 \vee v_1)$ |
| Red edges | $(u_2) \wedge (v_2) \wedge (\neg u_3 \vee \neg v_3) \wedge (\neg u_1 \vee \neg v_3) \wedge (\neg v_1 \vee \neg u_3) \wedge (u_1 \vee v_1)$ |

### 7.4.5 The complexity of $B_5^+$-colouring

**Case $B_5^+$-1 : No blue loops or blue loops on 1 or 2.** If there are no blue loops on $B_5$, the problem is NP-complete by restricting the input to edge-coloured graphs with only blue edges. If there is a blue loop on vertex 1 or 2, we can reduce the problem to a $T_5^+$-colouring problem using the Dominating Loop Lemma (Lemma 2.4.3).

**Case $B_5^+$-2: Blue loop on 0 or 3.** By symmetry we can assume there is a blue loop on 0. If there is a red loop on 1 or 2, we use the indicator $RB$ from above and produce a semicomplete directed graph with two directed cycles. Hence the problem is NP-complete. As before, adding more loops to this edge-coloured graph will not change the complexity.

Therefore, we can assume there is not a red loop on 1 nor 2. Let $I_1$ be $B$, the single blue edge on vertices $i_1$ and $j_1$. Let $I_2$ be $BRB$ the path on vertices $\{i_2, x, y, j_2\}$ with edges $i_2x$ and $yj_2$ blue and edge $xy$ red. The result of the indicator construction with respect to $(I_1, i_1, j_1), (I_2, i_2, j_2)$ has as a core a $T_4^+$ edge-coloured graph on vertices $\{0, 1, 2\}$. This is NP-complete. Therefore, all edge-coloured graphs in $B_5^+$-2 are NP-complete.

### 7.4.6 The complexity of $B_6^+$-colouring

**Case $B_6^+$-1: There are zero or one blue loops.** In the case there are no blue loops we can restrict the input to blue only and conclude the problem is NP-complete. Assume without loss of generality there is a blue loop on vertex 0. If any of $\{1, 2, 3\}$ do not have a red loop, say vertex 1, then the edge-coloured graph retracts to a $T_1^+$-colouring by mapping vertex 1 to vertex 0. If all three $\{1, 2, 3\}$ have red loops, then let $I_1$ be a single blue edge on vertices $i_1$ and $j_1$ with a red loop on each vertex. The result of the indicator construction with respect to $(I_1, i_1, j_1)$ is a monochromatic $K_3$. This problem is NP-complete.

**Case $B_6^+$-2: There are two or more blue loops.** Suppose without loss of generality that vertices 0 and 1 have blue loops. The $B_6^+$ retracts to the subgraph on $\{0, 2, 3\}$ by mapping vertex 1 to vertex 0. This is a $T_1^+$-colouring problem and is polynomial.

This completes the classification of all digon-free two-edge-coloured four-cliques.

## 7.5 Infinite families of polynomial problems

In this section we construct two infinite families of edge-coloured cliques for which the $H$-colouring problem is polynomial.

The first family begins with a single blue loop. Call this graph $H_1$. Given an integer $i \geq 2$, $H_i$ is constructed by adding a blue dominating loop (a vertex $v$ adjacent to all other vertices in blue together with a blue loop on itself) to $H_{i-1}$ in the case that $i$ is odd, and by adding a red dominating loop to $H_{i-1}$ in the case that $i$ is even. This is an infinite family of two-edge-coloured cliques. Call this family $\mathcal{H}$.

**Lemma 7.5.1** *For all $H_i \in \mathcal{H}$, $H_i$ is a core.*

**Proof.** Suppose there exists an $H_i$ that is not a core. Label the vertices of $H_i$ with $\{h_1, h_2, \ldots, h_i\}$ where $h_1$ is the original blue loop in $H_1$ and $h_j$ is the dominating loop added to $H_{j-1}$ to form $H_j$. Since $H_i$ is not a core, there exists a retraction of $H_i$ to a proper subgraph of itself. Suppose $h_j$ is mapped to $h_k$. All vertices with odd subscripts have blue loops and all vertices with even subscripts have red loops. Therefore $j$ and $k$ have the same parity. That is, a blue loop can not map to a red loop and vice versa. Hence, $|j-k| \geq 2$. Choose $m$ between $j$ and $k$ such that $m$ has different parity from $j$ and $k$. Suppose $j$ is odd, $m$ is even, and $k < m < j$. (All other cases are analogous.) The edge $h_j h_m$ is red and the edge $h_m h_k$ is blue. When $h_j$ is mapped to $h_k$, a red-blue digon is formed. This is a contradiction to the fact that $H_i$ contains no digons. ∎

**Theorem 7.5.2** *For all $H_i \in \mathcal{H}$, $H_i$-COL is polynomial.*

**Proof.** Let $H_i$ be a member of $\mathcal{H}$. Trivially, $H_1$-COL is polynomial. If $i > 1$, then $H_i$ has a dominating loop in red or blue. By the Dominating Loop Lemma (Lemma

2.4.3), $H_i$-COL polynomially transforms to $H_{i-1}$-COL. By induction, we can conclude $H_i$-colouring is polynomial for all $i$. ∎

The second family constructed is a family of loop-free cliques but on many edge-colours. Each edge-coloured graph also has only one edge between any pair of vertices. (That is, the edge-coloured graphs are digon-free.) The construction of the family is implicitly given in the proof of the next theorem.

**Theorem 7.5.3** *Let $k$ be a positive integer. For each $n$ such that $1 \leq n \leq 2^k$, there exists a loop-free edge-coloured clique, $H$, on $n$ vertices with multiplicity at most $k$ such that $H$-COL is polynomial. For all $n > 2^k$ and for all loop-free edge-coloured cliques, $H$, on $n$ vertices with multiplicity $k$, the $H$-COL problem is NP-complete.*

**Proof.** We prove the first part of the theorem by induction on $k$. If $k = 1$ then $n$ is either 1 or 2. A single vertex and a single blue edge are examples of polynomial graphs with multiplicity at most one. Suppose the statement is true for all $k \leq t$. Let $k = t + 1$. Choose $n$ such that $1 \leq n \leq 2^k$. If $n \leq 2^{k-1}$, by induction there is a polynomial graph without loops on $n$ vertices with $k - 1$ or fewer edge-colours. This edge-coloured graph satisfies the theorem. Therefore, we can assume $2^{k-1} < n \leq 2^k$. In particular, since $k > 1$, we have $n > 1$. We can partition $n$ into $n = n_1 + n_2$ such that $1 \leq n_1, n_2 \leq 2^{k-1}$. (For example $n_1 = 2^{k-1}$ and $n_2 = n - 2^{k-1}$ will work.) By induction we can find two loop-free edge-coloured cliques on $n_1$ and $n_2$ vertices respectively, such that each has multiplicity $k - 1$ and each is polynomial. Call these $H_1$ and $H_2$ respectively. Notice, we can choose the edge colours such that $H_1 \cup H_2$ has multiplicity $k - 1$. Suppose blue is not an edge-colour in $H_1$ or $H_2$. Let $H$ be the edge-coloured clique obtained by constructing the join of $H_1$ and $H_2$ with respect to blue. That is, add every edge between $H_1$ and $H_2$ in blue. By Lemma 3.5.1 we have that $H$-COL is polynomial. Also, $H$ has $k$ or fewer edge-colours.

To prove the second statement, let $H$ be any loop-free $n$-clique with multiplicity $k$ where $n > 2^k$. A result in [10] and [18] states that if the edges of an $n$-clique are coloured with $k$ colours and $n > 2^k$, then there exists a monochromatic odd cycle. Hence, $H$ must contain a monochromatic odd cycle. Since $H$ is loop-free, $H$-COL is NP-complete. ∎

# Chapter 8

# Back to One Edge-Colour

The results in this chapter concern problems for graphs and digraphs (multiplicity one). These results are stated in this context simply because the orginal questions asked were in this context or because it is unclear how to or even not possible to generalize the results to edge-coloured graphs.

## 8.1 Homomorphically Full Graphs

This section grew out of work on the Homomorphism Factoring Problem but is interesting in its own right. In this section we characterize those graphs that contain, as subgraphs, all of their homomorphic images. In fact, we give several characterizations of these graphs and in particular we show these graphs are perfect. We restrict our attention to loop-free graphs.

Recall that a homomorphism $f : G \to H$ which is both onto the vertices of $H$ and induces a mapping onto the edges of $H$ defines a *congruence*. Conversely, a congruence on $G$ implicitly defines a homomorphism from $G$ to the *quotient* of the congruence.

Let $G$ be a graph. Suppose $A$ and $B$ are two disjoint subsets of $V(G)$. We say $A$ is *adjacent* to $B$ is there exists $u \in A$ and there exists $v \in B$ such that $uv$ is an edge of $G$. We say a congruence $S_0, S_1, \ldots, S_k$ *induces a retraction* if for each $i$ there is $s_i \in S_i$ with the following property: "$S_i$ is adjacent to $S_j$ if and only if $s_i s_j \in E(G)$." That is, the subgraph induced by $\{s_i : 0 \leq i \leq k\}$ is the quotient of the congruence. Furthermore, the homomorphism $S_i \to s_i$, $0 \leq i \leq k$, induced by the congruence is the identity map on this subgraph. That is, the homomorphism induced by the congruence is a retraction and the quotient is a retract of $G$.

We now provide the central definition for this section. Given a graph $G$, we say $G$ is *homomorphically full* if every congruence on $V(G)$ induces a retraction. That is, given any congruence, $S_0, S_1, \ldots, S_k$ on $V(G)$, there exists $s_i \in S_i$ for each $i$ such that $S_i$ is adjacent to $S_j$ if and only if $s_i s_j \in E(G)$.

**Lemma 8.1.1** *Let $H$ be a homomorphically full graph. Every induced subgraph, $H'$, of $H$ is itself homomorphically full.*

**Proof.** Let $H'$ be an induced subgraph of $H$. Let $S_0, S_1, \ldots, S_k$ be a congruence on $H'$. Extend this congruence to $H$ be adding a class containing a single vertex for each vertex in $V(H) \backslash V(H')$. Let $S_0, S_1, \ldots, S_k, S_{k+1}, \ldots, S_K$ be the classes in this extended congruence. Since $H$ is homomorphically full, there exists $s_i \in S_i$ for all $1 \leq i \leq K$ such that $S_i$ adjacent to $S_j$ implies $s_i s_j \in E(H)$ for all $1 \leq i \leq j \leq K$. This statement is still true if we restrict $i$ and $j$ to the range $1 \leq i \leq j \leq k$. Moreover, since $H'$ is an induced subgraph for $1 \leq i \leq j \leq k$, $s_i s_j \in E(H)$ if and only if $s_i s_j \in E(H')$. Hence $H'$ is homomorphically full. ■

Recall that a retract of a graph $H$ is necessarily an induced subgraph of $H$, but the converse is not true. Consider the following example: $C_6$ is an induced subgraph of the 3-dimensional cube, $Q_3$, but there is no retraction of $Q_3$ to $C_6$. Furthermore, it is possible for a graph to be both a homomorphic image of $H$ and an induced subgraph of $H$, yet not a retract of $H$. For example, $C_6$ is both a homomorphic image and an induced subgraph of $Q_3 \cup P_6$ but it is not a retract of $Q_3 \cup P_6$.

We show below that given a graph $H$ such that every homomorphic image of $H$ is a subgraph of $H$, then $H$ is homomorphically full. To simplify the proof of our main theorem we begin with some preliminary results. We will then use these results to characterize homomorphically full graphs. We begin with a definition to help simplify the notation in the proofs. Let $H$ be a graph with vertex-set $\{u_0, u_1, \ldots, u_k\}$. Let $C$ be the congruence defined by:

$$\begin{aligned} S_0 &= \{u_0, u_1\} \\ S_i &= \{u_{i+1}\} \quad \text{for } 1 \leq i \leq k-1. \end{aligned}$$

The quotient of $C$, say $K$, is the graph that results when $u_0$ and $u_1$ are identified. Denote $K$ by $H_{u_0u_1}$.

**Lemma 8.1.2** *Let $H$ be a graph such that every homomorphic image is $H$ is a subgraph of $H$. Then $H$ has at most one nontrivial connected component.*

**Proof** Suppose not, and let $C_1$ and $C_2$ be distinct nontrivial connected components in $H$ so that $|V(C_1)| + |V(C_2)|$ is maximum. Suppose $u$ and $v$ are vertices of $H$ such that $u \in V(C_1)$ and $v \in V(C_2)$. It follows that $H_{uv}$ has a connected component of size $|V(C_1)| + |V(C_2)| - 1$, which is larger than any connected component of $H$. ∎

In view of this lemma, a homomorphically full graph may be assumed to be connected.

We make repeated use of the following argument. Let $F$ be a fixed graph. Suppose $H$ is a graph with a pair of non-adjacent vertices, say $u$ and $v$. Suppose there does not exist an induced copy of $F$ in $H$ that contains both $u$ and $v$, and there do not exist induced copies $F_1$ and $F_2$ of $F$ in $H$ such that $u \in V(F_1), v \in V(F_2)$ and $F_1 \backslash \{u\} = F_2 \backslash \{v\}$. Then every induced copy of $F$ in $H$ is still present in $H_{uv}$. Further, if the identification of $u$ and $v$ creates a new induced copy of $F$, then $H_{uv}$ contains more induced copies of $F$ than does $H$, and therefore can not be a subgraph of $H$.

**Lemma 8.1.3** *Suppose $H$ is a graph with the property that every homomorphic image of $H$ is a subgraph of $H$. Then $H$ has diameter at most two.*

**Proof** Suppose not, and let $x$ and $y$ be vertices with $d(x, y) = 3$. Since $x$ and $y$ have no common neighbours, there are no copies $F_1$ and $F_2$ of $K_3$ with $x \in V(F_1), y \in V(F_2)$ and $F_1 \backslash \{x\} = F_2 \backslash \{y\}$. Clearly there is no copy of $K_3$ in $H$ containing both $x$ and $y$. Thus every copy of $K_3$ in $H$ is still present in $H_{xy}$. Every path of length three from $x$ to $y$ creates a new copy of $K_3$ in $H_{xy}$. Hence,$H_{xy}$ contains more copies of $K_3$ than does $H$, and so it is not a subgraph, a contradiction. ∎

Given a graph $H$, we say two vertices $u$ and $v$ are *neighbourhood comparable* if either $N(u) \supseteq N(v)$ or $N(v) \supseteq N(u)$.

**Theorem 8.1.4** *Suppose $H$ is a graph such that every homomorphic image of $H$ is a subgraph of $H$. Then for all pairs $u$ and $v$ of non-adjacent vertices, $u$ and $v$ are neighbourhood comparable.*

**Proof.** Suppose the statement is false, and define $m$ to be the largest integer such that there exist non-adjacent vertices $a$ and $b$ with neither $N(a) \subseteq N(b)$ nor $N(b) \subseteq N(a)$ and an induced copy, say $Z$, of $K_{m-2}$ in $N(a) \cap N(b)$. Since $H$ has diameter two, the integer $m$ exists for all pairs $a$ and $b$ and is at least three. Let $u$ and $v$ be non-adjacent vertices for which $m$ is maximum, and let $x \in N(u) \backslash N(v)$ and $y \in N(v) \backslash N(u)$.

We show that $N(x) \supset V(Z)$. Since $xv \notin E(H)$, there is no copy of $K_m$ in $H$ that contains both $x$ and $v$. If there exist copies $F_1$ and $F_2$ of $K_m$ in $H$ such that $x \in V(F_1), v \in V(F_2)$ and $F_1 \backslash x = F_2 \backslash v$, then $x$ and $v$ belong to an induced copy of $K_{m+1} - e$. By the choice of $m$, and since $u \in N(x)$, this implies $N(x) \supset N(v)$ and, in particular, $N(x) \supset V(Z)$. On the other hand, if $F_1$ and $F_2$ do not exist, then every copy of $K_m$ in $H$ is still present in $H_{xv}$. The set $Z \cup \{u, v\}$ induces a copy of $K_m$ in $H_{xv}$. If this is a new copy, then $H_{xv}$ contains more copies of $K_m$ than does $H$, and therefore can not be a subgraph of $H$, contradicting our hypothesis. It follows that in $H$ the set $Z \cup \{u, v, x\}$ contains a copy of $K_m$, and as both $x$ and $u$ are non-adjacent to $v$, that $N(x) \supset V(Z)$. Similarly, $N(y) \supset V(Z)$.

First suppose that $xy \in E(G)$. Now $u$ and $v$ are non-adjacent; therefore, they do not belong to a common $K_{m+1}$. Nor are there copies $F_1$ containing $u$ and $F_2$ containing $v$ of $K_{m+1}$ such that $F_1 \backslash \{u\} = F_2 \backslash \{v\}$; otherwise, $m$ is not maximum. Hence, $H_{uv}$ contains more copies of $K_{m+1}$ than does $H$, a contradiction. Therefore, assume $xy \notin E(G)$.

Note that each of the pairs $\{u, v\}, \{x, v\}$, $\{u, y\}$ and $\{x, y\}$ has the property that the intersection of their neighbourhoods contains $Z$, they are non-adjacent, and in each pair, one vertex has a neighbour not adjacent to the other.

By the definition of $m$, there is no induced copy of $K_{m+1} - e$ in $H$ that contains $u$ and $v$. This follows the fact that $m$ is maximum over all such pairs. Suppose there do not exist induced copies $F_1$ and $F_2$ of $K_{m+1} - e$ in $H$ such that $u \in V(F_1), v \in V(F_2)$ and $F_1 \backslash \{u\} = F_2 \backslash \{v\}$. Then every induced copy of $K_{m+1} - e$ in $H$ is still present in $H_{uv}$. The graph $H_{uv}$ contains a copy of $K_{m+1} - e$ induced by $Z \cup \{u, x, y\}$. If this copy is new, then $H_{uv}$ contains more induced copies of $K_{m+1} - e$ than does $H$, a contradiction. It follows that this is not a new copy, and so $H$ must contain a copy of $K_{m+1} - e$, induced by an $(m+1)$-subset of the $(m+2)$-set $Z \cup \{u, v, x, y\}$. However, the removal of a single vertex from this $(m+2)$-set leaves two pairs of non-adjacent vertices and hence can not result in a $K_{m+1} - e$. The non-adjacent pairs of vertices are $\{u, v\}, \{x, v\}, \{u, y\}$, and $\{x, y\}$. Therefore the subgraphs $F_1$ and $F_2$ do exist.

Consider $F_1 \cap F_2$. Without loss of generality we can assume that $Z \subset F_1 \cap F_2$. Hence there are two more vertices, say $a$ and $b$, in $F_1 \cap F_2$. If $a$ and $b$ are non-adjacent, then $Z \cup \{a\}$ is a $K_{m-1}$ in the common neighbourhood of $u$ and $v$. This contradicts the choice of $m$. Hence, assume without loss of generality that $u$ and $a$ are non-adjacent. There are two cases to consider.

If $\{v, a\}$ is the non-adjacent pair of vertices in $F_2$, then $Z \cup \{b\}$ is a $K_{m-1}$ in the common neighbourhood of $u$ and $v$ contrary to the choice of $m$. Therefore $\{v, b\}$ is the non-adjacent pair of vertices in $F_2$. This implies $va$ is an edge.

Since $u$ and $v$ are not adjacent, there is no copy of $K_{m+1}$ in $H$ that contains $u$ and $v$. By the definition of $m$, there are no copies $G_1$ and $G_2$ of $K_{m+1}$ in $H$ such that $u \in V(G_1), v \in V(G_2)$ and $G_1 \backslash \{u\} = G_2 \backslash \{v\}$. Thus every copy of $K_{m+1}$ in $H$ is still present in $H_{uv}$. Moreover $H_{uv}$ contains a new copy of $K_{m+1}$ induced by $(V(F_1) \cup V(F_2))$, a contradiction. ■

We now characterize homomorphically full graphs.

**Theorem 8.1.5** *Suppose $H$ is a graph. The following statements are equivalent:*

*(a) The graph $H$ is homomorphically full.*

*(b) Every homomorphic image of $H$ is a retract of $H$.*

*(c) Every homomorphic image of $H$ is an induced subgraph of $H$.*

*(d) Every homomorphic image of $H$ is a subgraph of $H$.*

*(e) For any two non-adjacent vertices $u$ and $v$ of $H$ $u$ and $v$ are neighbourhood comparable.*

*(f) $H$ contains no induced $2K_2$ or $P_3$.*

**Proof.** (a) $\Rightarrow$ (b) In a homomorphically full graph the quotient of any congruence is a retract.

(b) $\Rightarrow$ (c) Every retract of $H$ is an induced subgraph of $H$.

(c) $\Rightarrow$ (d) Every induced subgraph is a subgraph.

(d) $\Rightarrow$ (e) This is Theorem 8.1.4.

(e) $\Rightarrow$ (a) Suppose we are given a congruence on $H$. Define a quasi-order on the vertices of $S_i$ by $u \geq v$ if and only if $N(u) \supseteq N(v)$ for all $t$. Since every pair of vertices in $S_i$ are neighbourhood comparable, every pair of vertices in $S_i$ are comparable under this order. Hence, there must be a maximal element under this order. Finally, any maximal element in each part will suffice as $s_i \in S_i$. Hence, $H$ is homomorphically full.

(e) $\Leftrightarrow$ (f) Suppose all pairs of non-adjacent vertices are neighbourhood comparable. Then $H$ contains no induced $2K_2$ or $P_3$ as both of these graphs contain a non-adjacent pair of vertices that are not neighbourhood comparable. On the other hand, suppose $H$ contains no induced $2K_2$ or $P_3$. Let $u$ and $v$ be pair of non-adjacent vertices. Suppose $u$ has a neighbour $x \notin N(v)$ and $v$ has a neighbour $y \notin N(u)$. Then subgraph induced by $\{u, x, y, v\}$ induces either a $2K_2$ or a $P_3$. ∎

**Corollary 8.1.6** *Every retract of a homomorphically full graph is itself homomorphically full.*

**Proof.** A retract of a graph is necessarily an induced subgraph. ∎

Complement reducible graphs (or cographs) are studied in [8]. Corneil, Lerchs, and Stewart-Burlingham show that cographs are perfect graphs and can be characterized as the graphs that contain no induced $P_3$. Hence, homomorphically full graphs are a subset of the cographs.

Our final result on homomorphically full graphs is that they are perfect. Since they are a subset of the cographs, this is immediate. However, we present a direct proof below.

**Theorem 8.1.7** *Every homomorphically full graph $H$ is perfect.*

**Proof.** Let $H = (V, E)$ be a homomorphically full graph. By Lemma 8.1.1 it suffices to prove that $\chi(H) = \omega(H)$. By definition of the chromatic number, $K_{\chi(H)}$ is a homomorphic image of $H$. Therefore, $K_{\chi(H)}$ is a subgraph of $H$, giving $\omega(H) \geq \chi(H)$. ■

## 8.2 The $H^k$-Colouring Problem

For the remainder of this chapter we make a slight change in nomenclature. In the digraph literature the word *colour* is used to refer to the vertices of $H$ in an $H$-colouring problem. This comes from the fact that $H$-colouring is a generalization of classical vertex-colouring. We have avoided the use of this term in this thesis so as to avoid confusion with edge-colours. However, for these final sections, the word *colour* will refer to the vertices of the target.

### 8.2.1 Powers of Oriented Paths

The following definition is taken from [14]. The definition is the main tool used in their algorithm.

**Definition.** Let $H$ be a directed graph and let $(v_1, v_2, \ldots, v_n)$ be an enumeration of its vertices. We say, a pair $(v_i, v_j)$ *dominates* a pair $(v_k, v_l)$, or $(v_i, v_j) \geq (v_k, v_l)$, if and only if $i \geq k$ and $j \geq l$. We say the pairs $(v_i, v_j)$ and $(v_k, v_l)$ are *crossing*, if and only if either ($i > k$ and $j < l$) or ($i < k$ and $j > l$). For pairs $(v_i, v_j)$ and $(v_k, v_l)$, the pair $(v_{min(i,k)}, v_{min(j,l)})$ is called the $\underline{X}$-pair of $(v_i, v_j)$ and $(v_k, v_l)$.

An enumeration of the vertices of $H$ is called an *$\underline{X}$-enumeration*, if for all pairs of edges $(v_i, v_j)$ and $(v_k, v_l)$ in $E(H)$, the $\underline{X}$-pair of $(v_i, v_j)$ and $(v_k, v_l)$ is in $E(H)$. The digraph $H$ has the $\underline{X}$-*property* (is an $\underline{X}$-digraph), if there exists an $\underline{X}$-enumeration of its vertices.

When $H$ is a digraph that has the $\underline{X}$-property, the algorithm presented in [14] solves H-COL in linear time. Let $H$ be a fixed oriented path. Since oriented paths have the $\underline{X}$-property, H-COL is polynomial. If we consider the complexity of colourings with powers of $H$, we find that the problem remains polynomial since $H^k$ has the $\underline{X}$-property as demonstrated below. In general this is not true. That is, given a graph with the $\underline{X}$-property, powers of the digraph do not necessarily have the $\underline{X}$-property.

**Lemma 8.2.1** *Let $H$ be an oriented path. The digraph $H^k$ has the $\underline{X}$-property.*

**Proof.** Let the vertices of $H$ be $\{0, 1, 2, \ldots, p\}$, where for $i \in \{0, 1, 2, \ldots, p-1\}$ either $(i, i+1) \in E(H)$ or $(i+1, i) \in E(H)$. Suppose $(i, j)$ and $(m, n)$ are two crossing arcs and without loss of generality further suppose $i < m$ and $j > n$.

If $i > n$, then we have $m > i > n$. Also, since $mn \in E(H^k)$ there must be a directed path of length at most $k$ from $m$ to $n$ in $H$. The vertex $i$ is between $m$ and $n$ and therefore it lies on the path. Hence, $in \in E(H^k)$. On the other hand, if $i < n$, then the directed path from $i$ to $j$ in $H$ passes through $n$. In this case, there is a directed path from $i$ to $n$ of length at most $k$. If $i = n$ then we have a directed path from $m$ to $i$ and a directed path from $i$ to $j$ with $i < m$ and $i = n < j$. This is impossible since the first path requires the arc $(i+1)(i)$ and the second path requires the arc $(i)(i+1)$. The path $H$ is an oriented path and can only contain one of these two arcs. Therefore the crossing pair's $\underline{X}$-pair is the arc $in$. ∎

**Theorem 8.2.2** *Let $H$ be an oriented path. Then the $H^k$-colouring problem is solvable in linear time.*

**Proof.** Use lemma 8.2.1 and the algorithm in [14].

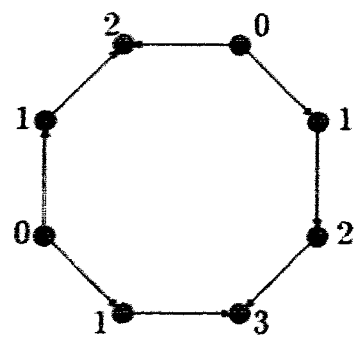


Figure 8.1: An oriented cycle that admits a homomorphism to $P_3^2$

Unfortunately, while the algorithm in [14] is elegant it does not give us much insight into which digraphs admit a homomorphism to $H$. This is also true when $H$ is an oriented path. This lack of a nice characterization perhaps indicates why the complexity of colouring by oriented paths proved elusive for some time.

### 8.2.2 Powers of Directed Paths

If we consider the case when $H$ is a directed path we are able to characterize the digraphs that admit a homomorphism to $H$. In particular, a digraph, $D$, admits a homomorphism to $H$ if and only if all cycles in $D$ are balanced and all paths in $D$ have net length less than $|H|$. See [16]. While it might be ambitious to try and characterize all digraphs that admit a homomorphism to some power of an oriented path, we can characterize all digraphs that admit a homomorphism to some power of a directed path.

Define $P_\infty$ to be the directed path with vertex-set the integers and $uv \in E(P_\infty)$ if and only if $v - u = 1$.

**Theorem 8.2.3** *Let $D$ be a digraph. Then $D \to P_\infty^k$ if and only for all cycles $C \to D$ we have $C \to P_\infty^k$.*

**Proof.** The necessity of the condition is obvious. If $C \to D$ and $D \to H^k$, then $C \to H^k$.

Let the colours of $H^k$ be $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$. Assume that all cycles that admit a homomorphism to $D$ also admit a homomorphism to $H^k$. Let $C$ be an oriented cycle. Consider $C$ as having its vertices on a circle. Each arc in $C$ will be either oriented clockwise or counterclockwise. The *net length* of $C$ is simply the absolute value of the number of clockwise oriented arcs in $C$ minus the number of counterclockwise oriented arcs in $C$. Of the two directions (clockwise and counterclockwise) call the arcs in one direction *forward* arcs and arcs in the other direction *backward* arcs. Choose these designations in such a way that $C$ has at least as many forward arcs as backward arcs. We will denote the number of each in $C$ as $forward(C)$ and $backward(C)$. For example, the cycle in Figure 8.1 has five forward arcs and three backward arcs.

**Claim 8.2.3.1** *$C \to P^k_\infty$ if and only if $forward(C) - k \cdot backward(C) \leq 0$.*

**Proof of Claim 8.2.3.1.** Suppose $C \to P^k_\infty$ under a homomorphism $f$. Each arc in $C$ will have its ends mapped to colours in $P^k_\infty$ such that the colours differ by at least one and by no more than $k$. Let the vertices of $C$ be labeled $c_0, c_1, \ldots, c_m$ so that the forward arcs are labeled $c_ic_{i+1}$ and the backward arcs are labeled $c_{i+1}c_i$. Indices are taken modulo $(m+1)$. Let $a$ be the arc in $C$ with end points $c_i$ and $c_{i+1}$. Note, $a$ may be oriented in either direction. Define $len(a) := f(c_{i+1}) - f(c_i)$. Since $f$ is a homomorphism it must be the case that $1 \leq |len(a)| \leq k$ for each arc, $a$, in $C$. By our choice of the labels for $C$ we also have $len(a) > 0$ when $a$ is a forward arc and $len(a) < 0$ when $a$ is a backward arc. Let $\mathcal{F}$ be the set of forward arcs in $C$ and $\mathcal{B}$ be the set of backward arcs in $C$. If we sum $len$ on the forward and backward arcs we get:

$$\begin{aligned} \sum_{a\in\mathcal{F}} len(a) &\geq forward(C) \\ \sum_{a\in\mathcal{B}} len(a) &\geq -k\cdot backward(C) \\ \sum_{a\in\mathcal{F}\cup\mathcal{B}} len(a) &= 0 \end{aligned}$$

The final summation is $f(c_1) - f(c_0) + f(c_2) - f(c_1) + \cdots + f(c_m) - f(c_{m-1}) + f(c_0) - f(c_m)$. Therefore, the final summation is zero. This implies $forward(C) - k \cdot backward(C) \leq 0$ and establishes the claim.

For an example, consider again the oriented cycle in Figure 8.1. The cycle has five forward arcs and three backward arcs. The numbers beside the vertices are the colours each is mapped to. In this case, we have a homomorphism to $P_3^2$.

We now assume that all the cycles in $D$ have this property. The strategy is to define a mapping on the vertices of $D$ to the vertices of $P_\infty^k$ and show this mapping is a homomorphism. We begin by defining a function $\psi$ from the Cartesian product of the oriented walks of $D$ crossed with the vertices of $P_\infty$ to the integers.

Let $W = w_0w_1\ldots w_m$ be an oriented walk in $D$. Assign to each pair $w_iw_{i+1}$ in $W$ a weight of $+1$ or $-k$. Assign $+1$ to the pair if $w_iw_{i+1}$ is an arc in $D$. Assign $-k$ to the pair if $w_{i+1}w_i$ is an arc in $D$. Since $P_\infty$ does not contain any cycles of length two, we can assume the same is true for $D$ and hence this assignment is well defined. Call this assigned weight $\omega(w_iw_{i+1})$. Define the function $\psi$ as follows. Let $c$ be any colour of $H$.

$$\psi(W,c) := c + \sum_{i=1}^{m-1} \omega(w_iw_{i+1})$$

In fact, $\psi(W,c)$ is the minimum colour that could be assigned to $w_m$ given $w_0$ has been coloured $c$.

We are now ready to define the homomorphism from $D$ to $P^k_\infty$. We define $l : V(D) \to V(P^k_\infty)$ as follows:

1. Arbitrarily map some vertex of D to 0. Call this vertex $Z$ (for zero).
2. Let $W$ be a walk from $Z$ to $u$. Define $l(u) := \max \psi(W,0)$, where this maximum is taken over all walks from $Z$ to $u$.

We need to show this maximum is well defined. This is potentially a problem since there are an infinite number of walks from $Z$ to $u$.

**Claim 8.2.3.2** *There exists a path $T$ from $Z$ to $u$ such that $\psi(T,0) = l(u)$.*

(Notice by proving this claim we will have proved $l$ is well defined since there are only finitely many paths from $Z$ to $u$.)

**Proof of Claim 8.2.3.2.** Let $W = (Z = w_0)w_1w_2\dots(w_m = u)$ be a walk from $Z$ to $u$. Suppose $W$ contains two vertices $x$ and $y$ such that $x = w_i$, $y = w_{i+1}$, and $x = w_{i+2}$. The weights $\omega(w_iw_{i+1})$ and $\omega(w_{i+1}w_{i+2})$ are $+1$ and $-k$ if $xy$ is an arc of $D$ and they are $-k$ and $+1$ if $yx$ is an arc of $D$. In either case, $\omega(w_iw_{i+1}) + \omega(w_{i+1}w_{i+2}) \leq 0$. Therefore, $\psi(W,0) \leq \psi(W',0)$ where $W' = w_0w_1\dots w_iw_{i+3}\dots w_m$. Hence, when maximizing $\psi$ we may restrict our attention to walks that do not contain a pair $\{x,y\}$ as above.

Suppose $W$ contains a cycle, $C$. Let $f = forward(C)$ and $b = backward(C)$. Our assumption on $D$ says $f - kb \leq 0$. Also $f \geq b$ by definition of $f$ and $b$ which implies

$-kf+b \leq 0$. If we let $C = c_0c_1 \ldots c_{r-1}c_0$, we have $\psi(C,j) \leq j$ by the fact that $f-kb$ and $-kf+b$ are nonpositive. Therefore, if we let $W''$ be the walk obtain by removing $C$ from $W$ we have $\psi(W,0) \leq \psi(W'',0)$.

Hence, if we consider all walks from $Z$ to 0, it must be the case that $\psi$ achieves its maximum on some path $T$. This establishes the claim.

Finally, we show $l$ is a homomorphism. Let $uv$ be an arc in $D$. Let $W$ be a walk from $Z$ to $u$ such that $l(u) = \psi(W,0)$. The walk $W'$ obtained by adding $v$ to the end of $W$ is a walk from $Z$ to $v$. Also $\psi(W',0) = l(u)+1$. Therefore, $l(v) \geq l(u)+1$. On the other hand, consider a walk $T$ from $Z$ to $v$ such that $\psi(T,0) = l(v)$. Let $T'$ be the walk obtained by adding $u$ to the end of $T$. This is a walk from $Z$ to $u$ such that $\psi(T',0) = l(v)-k$. In this case, $l(u) \geq l(v)-k$. Combining these we get $l(u)+k \geq l(v) \geq l(u)+1$. Therefore, $l(u)l(v)$ is an arc in $P^k_\infty$ and hence $l$ is a homomorphism. ∎

An immediate corollary to this theorem is a follows:

**Corollary 8.2.4** *Let $H$ be a digraph. Then the minimum integer $k$ for which $H \to P^k_\infty$ equals the maximum integer $m$ for which $m = \lceil forward(C)/backward(C) \rceil$ taken over all cycles $C$ in $H$.*

### 8.2.3 Powers of Directed Cycles

It is not surprising that the colouring problem for powers of directed paths are polynomial since the colouring problem for directed paths is polynomial. Also $P^{n-1}_n$-COL is polynomial since $P^{n-1}_n$ is a transitive tournament. See [1]. Conversely, $C^k_n$ is NP-complete for $k > 1$ by Theorem 5.2.4 in [25]. Perhaps what is surprising is that the problem of $G \to C_n^k$ remains NP-complete even when a homomorphism to a higher power of $C_n$ is provided as part of the instance. Formally we consider the following Restricted Homomorphism problem.

**RHP(H,Y)** (Restricted Homomorphism Problem)

INSTANCE: A directed graph $G$ and a homomorphism $G \to Y$.

QUESTION: Does there exist an H-colouring of G?

If we let $Y = C_n^l$ and $H = C_n^k$ we are asking if, given information about colouring $G$ with one power of $C_n$, can we colour $G$ with another power.

**Theorem 8.2.5** *The problem RHP($C_n^k$, $C_n^l$) is polynomial if $k = 1$ or $k \geq l$. Otherwise RHP($C_n^k$, $C_n^l$) is NP-complete.*

**Proof.** If $k = 1$, then RHP($C_n^k$, $C_n^l$) is polynomial since colouring by a directed cycle is polynomial. If $k \geq l$, then RHP($C_n^k$, $C_n^l$) is polynomial since $C_n^l \to C_n^k$ and $G \to C_n^l$.

Therefore, suppose $1 < k < l \leq n-1$. By the results in [26] we know RHP($H^*$,$Y^*$) $\alpha$ RHP($H$,$Y$), where the indicator $(J, x_1, w)$ is constructed as follows: Let $X$ be the transitive tournament on $k+2$ vertices. (Note that $X \not\to C_n^k$, but $X \to C_n^l$. To see this, let the vertices of $C_n$ be $\{0, 1, 2, \ldots, n-1\}$ where $ij \in E(C_n)$ if and only if $j - i \equiv 1 \bmod n$. Since $X$ is a core, $X \to C_n^l$ if and only if $X$ is a subgraph of $C_n^l$. The set of vertices $\{0, 1, 2, \ldots, k+1\}$ induces a subgraph isomorphic to $X$ since $l > k$. There is no subgraph of $C_n^k$ isomorphic to $X$ since the source in $X$ has outdegree $k+1$ and all the vertices in $C_n^k$ have outdegree $k$.)

Let the vertices of $X$ be labelled $x_0, x_1, \ldots, x_{k+1}$ where $x_i x_j$ is an arc if and only if $i < j$. Let $J$ be the graph constructed by removing the arc $x_0 x_1$ from $X$ and adding a

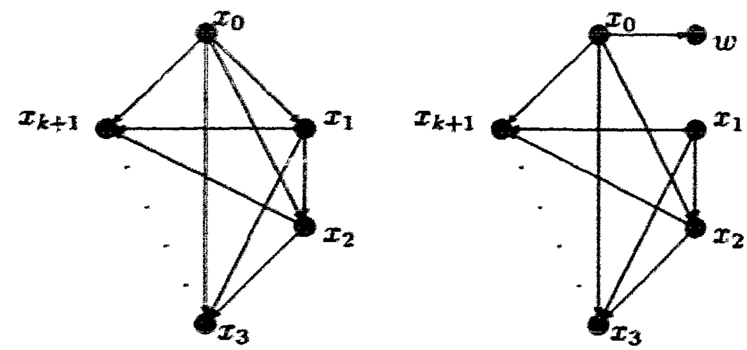


Figure 8.2: The digraphs $X$ and $J$

new vertex $w$ together with the arc $x_0w$. We now use the indicator $(J, x_1, w)$. We need two observations. First, $J \to C_n^k$ but in any such homomorphism $x_1$ and $w$ receive different colours. If $x_1$ and $w$ receive the same colour, then there is a homomorphism from $X$ to $C_n^k$ contrary to our above discussion. To see $J \to C_n^k$, identify $x_0$ and $x_1$. Call this vertex $x_0$. This produces a transitive tournament on $k+1$ vertices with an extra arc, $x_0w$, incident with the source. This digraph admits a homomorphism to the transitive tournament on $k+1$ vertices by identifying $w$ with any out neighbour of vertex $x_0$. Recall that $k > 1$ and therefore $x_0$ has at least one out-neighbour.

In $C_n^k$, each vertex is the source of a transitive tournament on $k+1$ vertices. Therefore, each vertex in $C_n^k$ can be the image of $x_0$ in the mapping of $J \to C_n^k$. Since $w$ can be mapped to any out neighbour of $x_0$, we see that $H^* = H$.

Identifying $x_1$ and $w$ produces a transitive tournament on $k+2$ vertices and this is a subgraph of $C_n^l$. This means that $J \to C_n^l$ in such a way that $x_1$ and $w$ receive the same colour. This implies that $C_n^{l^*}$ has a loop. Therefore, any digraph trivially admits a homomorphism to $C_n^{l^*}$. Hence, RHP($H^*$,$Y^*$) is equivalent to $H^*$-COL. In particular, RHP($C_n^k$, $C_n^l$) is NP-complete for $1 < k < l$. ∎

### 8.2.4 Powers of Undirected Graphs

To complete this section we consider the $H^k$-colouring problem for undirected graphs $H$. In general $H$-colouring is NP-complete whenever $H$ contains an odd cycle [19]. However, we can consider a restricted homomorphism problem as we did in the case of directed cycles. We use the following definition and theorem from [26].

RHP($H$,$Y$) (Restricted Homomorphism Problem)

INSTANCE: A graph $G$ and a homomorphism $G \to Y$.

QUESTION: Doest there exist a homomorphism $G \to H$?

We also use the notation $\omega(H)$ to denoted the size of the largest clique in $H$.

**Theorem 8.2.6 (MacGillivray)** *If there is an $(\omega(H)+1)$-critical subgraph of $Y$ (a subgraph whose chromatic number is $(\omega(H)+1)$ but whose proper subgraphs all have chromatic number smaller than $(\omega(H)+1)$) that is not contained in $H$, then RHP($H$,$Y$) is NP-complete.*

We use Theorem 8.2.6 to show that $H^k$-colouring is NP-complete even when a homomorphism to a larger power is provided as part of the instance, except when the problem is clearly polynomial.

**Theorem 8.2.7** *Let $H$ be a graph and $l$ and $k$, $l < k$, be two integers. Then RHP($H^l$,$H^k$) is NP-complete if $H^l \neq H^k$ and is polynomial otherwise.*

**Proof.** We prove the latter statement first. If $H^l = H^k$, then the homomorphism, $G \to H^k$, provided in the instance makes the answer YES.

**Claim 8.2.7.1** *Let $H$ be a connected graph and $l$ and $k$, $l < k$, be two integers. Then $\omega(H^l) < \omega(H^k)$ if and only if $H^l \neq H^k$.*

**Proof of Claim 8.2.7.1.** The necessity of the condition is obvious. If $H^l = H^k$, then $\omega(H^l) = \omega(H^k)$.

If $H^l \neq H^k$, then by definition of $H^k$, $H^l$ is a proper subgraph of $H^k$. This immediately gives us that $H^l$ is not a clique. Let $X$ be a maximum clique in $H^l$. Let $v$ be a vertex in $X$ and $u$ be a vertex not in $X$. Because $H$ is connected, there must be a path $(v = p_0)p_1p_2\ldots(p_t = u)$ in $H$ connecting the two vertices. Let $w$ be the first vertex in this path not in $X$. The vertex $w$ is distance one in $H$ from some vertex in $X$. Each pair of vertices in $X$ are at most distance $l$ apart in $H$. Therefore, $w$ is at most distance $l + 1$ from each vertex in $X$. Hence, $X \cup \{w\}$ is a clique in $H^{l+1}$. We conclude

$$\omega(H^l) < \omega(H^{l+1}) \leq \omega(H^k)$$

This establishes Claim 8.2.7.1

Since we have established that the problem is polynomial when $H^k = H^l$, we consider the case when $H^l \neq H^k$. In light of Theorem 3.1 in [2] we can restrict our attention to the case when $H$ is a connected core when trying to show $H^k$-colouring is NP-complete. This allows us to use the above claim. The subgraph $X \cup \{w\}$ described above is an $(\omega(H^l) + 1)$-critical subgraph of $H^k$. By Theorem 8.2.6, RHP($H^k$,$H^l$) is NP-complete. ∎

## 8.3 Homomorphisms to $H^\infty$

When considering powers of a directed graph we see that eventually successive powers are equal. That is, let $k$ be the greatest distance between any pair of points in $H$. Then

$$H^k = H^{k+1} = H^{k+2} = \cdots$$

For convenience we will call this digraph $H^\infty$. The digraph is transitively closed. Let $uv$ and $vw$ be two arcs in $H^\infty$. This implies the arc $uw$ is in $H^\infty$.

**Theorem 8.3.1** *Let $H$ be an acyclic digraph. Then the $H^\infty$ colouring problem is polynomial.*

**Proof.** We define a function on $V(H^\infty)$ that is a retraction to the largest transitive tournament in $H^\infty$. Let $u$ be a vertex in $H^\infty$. Define $f(u)$ to be the size of the largest transitive tournament of which $u$ is the sink. Let $uv$ be an arc in $H^\infty$ and let $X$ be a transitive tournament with $u$ as a sink. Then $X \cup \{v\}$ is a transitive tournament with $v$ as a sink since $H^\infty$ is transitively closed. Hence $f(u) < f(v)$. Also, the largest value any vertex can be assigned is certainly less than or equal to the size of a maximum transitive tournament in $H^\infty$. If we label the vertices of some maximum transitive tournament in $H^\infty$ with the values $\{0, 1, 2, \ldots, t\}$, we see that $f$ is a retraction to this maximum transitive tournament. Hence, the core of $H^\infty$ is a transitive tournament. The $H$-colouring problem for transitive tournaments is polynomial, [1]. ∎

We require the following definition in the next proof. Let $H$ be a digraph. The graph *undir(H)* is the graph with vertex-set $V(H)$ and edge set $uv \in E(undir(H))$ if and only if both $uv$ and $vu$ are arcs in $H$. In is straightforward to check that if $H$ is a digraph such that $undir(H)$-COL is NP-complete, then $H$-COL is NP-complete.

**Theorem 8.3.2** *Let $H$ be a digraph containing a strong component of size at least three. Then the $H^\infty$ colouring problem is NP-complete.*

**Proof.** Let $C$ be a strong component in $H$ of length at least three. Let $u$ and $v$ be a pair of vertices in $C$. Both $uv$ and $vu$ are arcs in $H^\infty$. So if we look at *undir*$(H^\infty)$, $C$ is a clique of size at least three. Therefore, the problem is NP complete. ∎

**Corollary 8.3.3** *Let $H$ be a digraph containing a directed cycle of size at least three. Then the $H^\infty$ colouring problem is NP-complete.*

**Theorem 8.3.4** *Let $H$ be a digraph containing two directed two-cycles $C_1$ and $C_2$ such that there is a directed path from a vertex in $C_1$ to a vertex in $C_2$. Then $H^\infty$-COL is NP-complete.*

**Proof.** We can order the strong components of $H^\infty$ so that given two strong components $T_1$ and $T_2$ all the arcs between $T_1$ and $T_2$ are oriented towards $T_2$ if and only if $T_1$ is less that $T_2$ in this ordering. Choose $C_1$ (respectively $C_2$) to be the first (respectively second) two-cycle in this ordering. Let $w$ be one of the vertices of $C_2$.

Define $S$ to be the digraph on vertices $\{0,1,2\}$ such that $\{0,1\}$ is a two-cycle and $(1,2)$ is an arc. Let $k_1$ be 2. Let $h_1$ be a two vertex in $C_1$. Let $j$ be the vertex 0 in $S$. Then subindicator construction with respect to $(S, h_1, j)$ on $H^\infty$ produces the digraph consisting of $C_1$, $C_2$ and all arcs from $C_1$ to $C_2$. This homomorphism problem is NP-complete by Theorem 3.6 from [2]. ∎

It remains to consider those graphs in the class where no pair of two-cycles is joined by a directed path. We can construct infinitely many NP-complete examples of digraphs in this class. Let $G_1$ and $G_2$ be obtained from a transitive tournament on five vertices by adding the arcs $(3,2)$ and $(4,3)$, respectively. Construct $I$ from $G_1$ and $G_2$ by identifying them at vertex 5 (the sink of each). Then $I$ is a core. Let $u$ be vertex 1 of $G_1$ in $I$, and $v$ be vertex 1 of $G_2$ in $I$. Let $G$ be a digraph for which $G$-col is NP-complete. Let $^*G$ be the result of substituting $(I,u,v)$ for each edge of $G$. Then $^*G^\infty =^* G$. The result of applying the indicator construction with respect to $(I,u,v)$ to $^*G$ is $G$, and $G$-col is NP-complete by hypothesis. Thus some digraphs in this class are NP-complete, and some are polynomial.

# Bibliography


[1] J. Bang-Jensen, P. Hell, and G. MacGillivray, The complexity of colouring by semicomplete digraphs. *SIAM Journal on Discrete Mathematics.* **1**(1988) pp 281-298.

[2] J. Bang-Jensen and P. Hell, The effect of two cycles on the complexity of colourings by direct graphs. *Discrete Applied Mathematics.* **26**(1990) pp 1-23.

[3] J. Bang-Jensen, P. Hell, and G. MacGillivray, On the complexity of colouring by superdigraphs of bipartite graphs, *Discrete Mathematics.*, to appear.

[4] J. Bang-Jensen, P. Hell, and G. MacGillivray, Hereditarily hard colouring problems, submitted to *Journal of Computing Systems Science.*

[5] J. A. Bondy and U. S. R. Murty, "*Graph Theory with Applications.*" Elsevier Press, New York, 1976.

[6] R. C. Brewster and G. MacGillivray, A note on restricted H-colourings. *Information Processing Letters* **40**(1991) pp 149-151.

[7] R. C. Brewster and G. MacGillivray, Homomorphically full graphs, *in preparation.*

[8] D. G. Corneil, H. Lerchs, and L. Stewart Burlingham, Complement reducible graphs. *Discrete Applied Mathematics* **3**(1981) pp 163-174

[9] S. Even, A. Itai, and A. Shamir, On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing* **5**(1976) pp 691-703.

[10] P. Erdős and R. L. Graham, On partition theorems for finite graphs, *Infinite and finite sets* (Colloq., Keszthely, 1973), edited by A. Hajnal, R. Rado, and V. T. Sós, Colloq. Math. Soc. J. Bolyai, **10**, North-Holland Publishing Co., Amsterdam, 1975, 515-527.

[11] Z. Furedi, J. R. Griggs, and D. J. Kleitman, Pair labelings with given distance, *SIAM Journal on Discrete Mathematics* **2**(1989), 491-499.

[12] M. R. Garey and D. S. Johnson, "*Computers and Intractability*", Freeman, N.Y. 1979

[13] J. R. Griggs and R. K. Yeh, Labelling graphs with a condition at distance two. *SIAM Journal on Discrete Mathematics* (1991), to appear.

[14] W. Gutjahr, E. Welzl, and G. Woeginger, Polynomial graph-colorings. *University of Berlin Technical report B-88-06.*

[15] W. Gutjahr, Graph Colorings, Ph. D. Thesis, Freie Universität Berlin, 1991.

[16] R. Häggkvist, P. Hell, D. J. Miller, and V. Neumann-Lara, On multiplicative graphs and the product conjecture, *Combinatorica* **8**(1988), 71-81.

[17] W. K. Hale, Frequency assignment: theory and applications. *Proceedings of the IEEE* **68**(1980), 1497-1514.

[18] F. Harary, D. Hsu, and Z. Miller, The biparticity of a graph, *Journal of Graph Theory* **1**(1977), 131-133.

[19] P. Hell and J. Nešetřil, On the complexity of $H$-colouring. *Journal of Combinatorial Theory, Series B.* **48**(1990) pp 92-110.

[20] P. Hell and X. Zhu, Homomorphisms to oriented paths. *In preparation.*

[21] P. Hell, H. Zhou, and X. Zhu, Homomorphisms to oriented cycles, *Combinatorica*, to appear.

[22] P. Hell, X. Zhu, The existence of homomorphisms to oriented cycles, submitted to *SIAM Journal on Discrete Math.*

[23] F. T. Lenighton, A graph colouring algorithm for large scheduling problems, *J. Res. Nat. B. Standards* **84**(1979), 489-496.

[24] G. MacGillivray, On the complexity of colourings by vertex-transitive and arc-transitive digraphs. *SIAM Journal on Discrete Mathematics* **4**(1991), 397-408.

[25] G. MacGillivray, Generalized colourings. Ph. D. Thesis. Simon Fraser University, 1989.

[26] G. MacGillivray, Notes on restricted H-colourings. Personal Communication.

[27] H. A. Maurer, J. H. Sudborough, and E. Welzl, On the complexity of the general coloring problem. *Information and Control.* **51**(1981), 123-145.

[28] D. J. Miller, The categorical products of graphs. *Canadian Journal of Mathematics.* **20**(1968). pp 1511-1521.

[29] T. J. Schaefer, The complexity of satisfiability problems. *Proc. 10th Ann. ACM Symp. on Theory of Computing,* Association for Computing Machinery, New York, pp 216-226.

[30] X. Zhu, A polynomial algorithm for homomorphisms to oriented cycles. Manuscript, 1991.